

UNIVERSIDADE DE LISBOA  
FACULDADE DE CIÊNCIAS  
DEPARTAMENTO DE INFORMÁTICA



## **Aplicação de Tratamento Automatizado de Faturas**

Gustavo José Pires da Silveira e Lorena

**Mestrado em Informática**

Trabalho de Projeto orientado por:  
Professor Doutor Thibault Nicolas Langlois







*Para a minha mãe*



## ***RESUMO***

As empresas cada vez mais procuram automatizar os seus processos, em especial os que incluem atividades de natureza rotineira, repetitiva e que envolvem um elevado volume de dados a serem manuseados. É objetivo das organizações melhorar a eficiência e eficácia organizacional, diminuindo tempos e custos de funcionamento associados.

As plataformas de computação na nuvem, como a Amazon, têm desempenhado um papel essencial nesta transformação e cada vez mais dispõem de uma diversidade de serviços que permitem solucionar problemas que, de outra forma, exigiria recursos significativos que são inexistentes ou escassos.

Este projeto foi realizado na Management Solutions e teve como objetivo desenvolver uma aplicação que permitisse melhorar o processo de tratamento e análise de faturas de despesas dos colaboradores. Pretendeu-se utilizar algumas ferramentas a que todos os colaboradores acedem com o telemóvel para fazer o input da fatura através de uma imagem que, posteriormente fosse processada de forma mais automatizada possível, para extração dos vários campos que a integram.

Para este efeito, foram equacionadas duas abordagens, uma que pudesse dar resposta imediata ao problema e que teve por base a construção de algoritmos de procura específicos e outra que se suportasse num modelo de aprendizagem profunda com um dataset que seria continuamente atualizado com as faturas que viessem a ser introduzidas ao longo do tempo pelos colaboradores.

As duas soluções revelaram-se válidas, a primeira é uma solução transitória, até que a segunda solução se apresente com níveis de robustez suficientes para ser utilizada com melhores resultados do que a primeira, ou seja, exista informação suficiente que permita ao modelo de aprendizagem profunda ser eficiente e preciso nas respostas.

**Palavras-chave:** Computação na nuvem, aprendizagem profunda, tratamento e análise de faturas de despesas.

## ***Abstract***

Companies are increasingly seeking to automate their processes, especially those that include routine and repetitive nature activities and involving a high volume of data to be handled. The goal is to improve organizational efficiency and effectiveness, decreasing associated operating times and costs.

Cloud computing platforms such as Amazon have played a key role in this transformation and have a variety of services to solve problems that would otherwise require significant resources that are non-existent or scarce.

This project was carried out at Management Solutions and aimed to develop an application that would improve the process of treatment and analyzing employee expenses invoices. It was intended to use some tools that all employees access as the mobile phone to make the invoice input through an image that would later be processed in the most automated possible way, to extract the various fields that integrate it.

For this purpose, two approaches were considered, one that could provide an immediate response to the problem and which was based on the construction of specific search algorithms and another that was supported on a deep learning model with a dataset that would be continuously updated with the invoices that would be introduced over time by employees.

Both solutions proved to be valid, the first is a transitory solution, until the second solution presents sufficient levels of robustness to be used with better results than the first one, meaning that, there is enough information to allow the deep learning model to be efficient and precise in the answers.

**Keywords:** Cloud computing, deep learning, treatment and analyzing expenses invoices.



# Contéudo

|            |  |    |
|------------|--|----|
| Capítulo 1 | Introdução .....   | 1  |
| 1.1        | Motivação .....  | 1  |
| 1.2        | Objetivos .....  | 2  |
| 1.3        | Metodologia .....  | 3  |
| 1.4        | Enquadramento Institucional .....  | 4  |
| 1.5        | Organização do Documento .....   | 4  |
| Capítulo 2 | Estado da Arte .....   | 6  |
| 2.1        | Computação na Nuvem .....  | 7  |
| 2.1.1      | Enquadramento .....  | 7  |
| 2.1.2      | Amazon Web Services .....  | 8  |
| 2.1.3      | Google Cloud Platform .....  | 10 |
| 2.1.4      | Azure .....  | 11 |
| 2.1.5      | Sumarização das plataformas de nuvem .....   | 12 |
| 2.2        | Técnicas Analisadas para a Aplicação .....   | 13 |
| 2.2.1      | Extração de Texto .....  | 13 |
| 2.2.2      | Tradução .....   | 14 |
| 2.2.3      | Execução do código – Armazenamento da Aplicação .....                                  | 14 |
| 2.2.4      | Pesquisa de campos com NLP .....   | 16 |
| 2.2.5      | API para Conexão de Sistemas .....   | 17 |
| 2.2.6      | Serviço de Armazenamento de Dados (Big Data) .....                                     | 18 |
| 2.2.7      | Base de Dados – NoSQL .....  | 19 |
| 2.2.8      | Serviços de Deep Learning para reconhecer imagens .....                                | 20 |
| 2.2.9      | Sistematização das Técnicas Analisadas .....   | 21 |
| 2.3        | Natural Language Processing .....  | 21 |
| 2.4        | Deep Learning .....  | 23 |
| 2.4.1      | Rede Neural Profunda .....   | 24 |
| 2.4.2      | Rede Neural Convolucional .....  | 25 |
| Capítulo 3 | Desenvolvimento do Projeto de Tratamento de Faturas .....                              | 27 |
| 3.1        | Diagnóstico .....  | 27 |
| 3.1.1      | Tipologia de Faturas .....   | 28 |
| 3.1.2      | Possíveis Soluções para tratamento de faturas .....                                    | 30 |
| 3.2        | Seleção da Plataforma .....  | 31 |
| 3.3        | Especificações da Aplicação .....  | 31 |
| 3.4        | Diagrama de processos da solução com algoritmos de pesquisa de campos de faturas ..... | 32 |

|            |  |    |
|------------|--|----|
| 3.4.1      | Envio da foto para processamento .....                                     | 33 |
| 3.4.2      | Validação da qualidade da fatura .....                                     | 34 |
| 3.4.3      | Retorno quando há má qualidade da imagem .....                             | 35 |
| 3.4.4      | Extração do texto da fatura .....  | 36 |
| 3.4.5      | Tratamento dos dados e procura de campos .....                             | 36 |
| 3.4.6      | Dados Armazenados .....  | 42 |
| 3.4.7      | Retorno Devolvido pelos Algoritmos Desenvolvidos.....                      | 43 |
| 3.4.8      | Organização dos Dados Corretos Finais numa Base de Dados.....              | 44 |
| 3.5        | Diagrama de Processos da Solução com Modelo de Deep Learning .....         | 44 |
| 3.5.1      | Identificação das coordenadas da imagem – Dataset e Treino do Modelo ..... | 46 |
| 3.5.2      | Retorno das coordenadas e processamento dos algoritmos .....               | 51 |
| Capítulo 4 | Resultados e Avaliações.....   | 54 |
| 4.1        | Resultados do modelo para identificar tipologia de fatura.....             | 54 |
| 4.2        | Resultados da solução com algoritmos de procura .....                      | 55 |
| 4.3        | Resultados da solução com modelo de Deep Learning.....                     | 60 |
| Capítulo 5 | Conclusões e Trabalho Futuro .....   | 67 |
| 5.1        | Conclusões .....   | 67 |
| 5.2        | Trabalhos Futuros .....  | 69 |

# Lista de Figuras

|   |    |
|---|----|
| Figura 1.1: Metodologia.....  | 3  |
| Figura 2.1: Características necessárias para o desenvolvimento da aplicação.....  | 6  |
| Figura 2.2: Representação de Word Embedding no espaço.....  | 22 |
| Figura 2.3: Representação das técnicas: Inteligência Artificial, Aprendizagem Automática e Aprendizagem Profunda. (Chollet,2018) .....  | 23 |
| Figura 2.4: Rede neural feed-forward multi-camada. (Chollet,2018).....  | 24 |
| Figura 2.5: Uma arquitetura CNN comprimida com cinco camadas. (Chollet,2018).....   | 25 |
| Figura 3.1: Exemplo da primeira fatura com todos os campos .....  | 29 |
| Figura 3.2: Exemplo da segunda fatura com quase todos os campos presentes .....   | 29 |
| Figura 3.3 Exemplo da terceira fatura com poucos campos especificados .....   | 30 |
| Figura 3.4 Diagrama de Processos da solução com algoritmos de pesquisa de campos de faturas .....   | 32 |
| Figura 3.5: Vista do utilizador antes do processamento de uma fatura.....   | 33 |
| Figura 3.6: O kernel de Laplace.....  | 34 |
| Figura 3.7 Foto retornada com o valor da variância de Laplace .....   | 35 |
| Figura 3.8 Foto considerada como desfocada.....   | 35 |
| Figura 3.9 : Solução inicial equacionada para descobrir os campos necessários.....  | 36 |
| Figura 3.10: Algoritmo desenvolvido para descobrir o valor total de uma fatura. ....  | 37 |
| Figura 3.11: Algoritmo desenvolvido para descobrir a data de uma fatura. ....   | 38 |
| Figura 3.12: Algoritmo desenvolvido para descobrir a data de uma fatura. ....   | 39 |
| Figura 3.13: Algoritmo desenvolvido para descobrir o país da fatura. ....   | 39 |
| Figura 3.14: Algoritmo desenvolvido para identificar a moeda da fatura. ....  | 40 |
| Figura 3.15: Algoritmo desenvolvido para converter o a moeda da fatura para euros. ....   | 40 |
| Figura 3.16: Algoritmo desenvolvido para descobrir o Tipo Impositivo (% específica por produto), Base Tributável e IVA da fatura (global e parcial). ....                           | 41 |
| Figura 3.17: Algoritmo desenvolvido para descobrir o número fiscal de uma fatura.....   | 41 |
| Figura 3.18: Algoritmo desenvolvido para descobrir a Empresa.....   | 41 |
| Figura 3.19 Hierarquia dos campos a encontrar numa fatura .....   | 42 |
| Figura 3.20: Output de uma fatura com os dados encontrados e fatura correspondente.....   | 43 |
| Figura 3.21 Diagrama de Processos da solução com Deep Learning .....  | 45 |
| Figura 3.22: Representação das coordenadas consideradas existentes numa fatura .....  | 47 |
| Figura 3.23: Representação das coordenadas que o Amazon Textract identifica para cada texto encontrado.....   | 48 |
| Figura 3.24: Processo para o desenvolvimento do Dataset gerado automaticamente.....   | 49 |
| Figura 3.25: Coordenadas retornadas pelo modelo de Deep Learning desenvolvido por uma fatura da Uber .....  | 51 |
| Figura 3.26: Correspondência entre as coordenadas enviadas pelo modelo e as coordenadas identificadas pelo Amazon Textract retornando o texto que apresenta a menor distância ..... | 52 |
| Figura 3.27 Representação do algoritmo que faz a validação dos dados que são retornados pelo modelo de DL .....   | 53 |
| Figura 4.1: Três faturas com o respetivo Tipo identificado pelo modelo de ML .....  | 55 |
| Figura 4.2: Primeira fatura com os respetivos resultados obtidos pela solução que utiliza algoritmos de procura pelos campos.....   | 56 |
| Figura 4.3 Segunda fatura com os respetivos resultados obtidos pela solução que utiliza algoritmos de procura pelos campos.....   | 57 |
| Figura 4.4 Terceira fatura com os respetivos resultados obtidos pela solução que utiliza algoritmos de procura pelos campos.....  | 58 |

|   |    |
|---|----|
| Figura 4.5 Quarta fatura com os respetivos resultados obtidos pela solução que utiliza algoritmos de procura pelos campos.....        | 58 |
| Figura 4.6: Gráfico que identifica a perda total do modelo treinado em DL .....   | 61 |
| Figura 4.7: Primeira fatura com os respetivos resultados obtidos pela solução que utiliza o modelo de DL pertencente ao grupo 1 ..... | 62 |
| Figura 4.8 Segunda fatura com os respetivos resultados obtidos pela solução que utiliza o modelo de DL pertencente ao grupo 1 .....   | 63 |
| Figura 4.9 Primeira fatura com os respetivos resultados obtidos pela solução que utiliza o modelo de DL pertencente ao grupo 2.....   | 63 |
| Figura 4.10 Segunda fatura com os respetivos resultados obtidos pela solução que utiliza o modelo de DL pertencente ao grupo 2.....   | 64 |
| Figura 4.11 Primeira fatura com os respetivos resultados obtidos pela solução que utiliza o modelo de DL pertencente ao grupo 3.....  | 64 |
| Figura 4.12 Segunda fatura com os respetivos resultados obtidos pela solução que utiliza o modelo de DL pertencente ao grupo 3.....   | 65 |

## Lista de Tabelas

|   |    |
|---|----|
| Tabela 2.1 Pontos fortes e fracos das várias plataformas de nuvem .....   | 12 |
| Tabela 2.2 Sistematização das funcionalidades necessárias para a aplicação apresentadas pelos principais serviços de nuvem existentes no mercado..... | 21 |
| Tabela 3.1 Campos específicos para extração.....  | 28 |
| Tabela 3.2 Representação dos dados que ficam agrupados na Base de Dados Dynamo .....  | 44 |
| Tabela 4.1 Resultados apresentados pelo modelo desenvolvido para descobrir o tipo de fatura.....  | 54 |
| Tabela 4.2 Resultados obtidos num conjunto de 200 faturas espanholas. ....  | 59 |
| Tabela 4.3 Resultados da precisão com recurso ao modelo de DL.....  | 65 |

## Lista de Acrónimos

| Acrónimos | Designação                         |
|-----------|------------------------------------|
| ML        | Aprendizagem Automática            |
| DL        | Aprendizagem Profunda              |
| CNN       | Rede Neural Convolucional          |
| NLP       | Linguagem Natural de Processamento |
| CPU       | Unidade Central de Processamento   |
| GPU       | Unidade de Processamento Gráfico   |
| OCR       | Reconhecimento Ótico de Caracteres |
| MS        | Management Solutions               |
| VPN       | Redes Virtual Privada              |
| SaaS      | Software como um Serviço           |
| PaaS      | Plataforma como um Serviço         |
| IaaS      | Infraestrutura como um Serviço     |
| GCP       | Google Cloud Platform              |
| AWS       | Amazon Web Services                |

## List of Acronyms

| Abbreviations | Explanation                   |
|---------------|-------------------------------|
| ML            | Machine Learning              |
| DL            | Deep Learning                 |
| CNN           | Convolutional Neural Network  |
| NLP           | Natural Language Processing   |
| CPU           | Central Process Unit          |
| GPU           | Graphics Processing Unit      |
| OCR           | Optical Character Recognition |
| MS            | Management Solutions          |
| VPN           | Virtual Private Network       |
| SaaS          | Software as a Service         |
| PaaS          | Platform as a Service         |
| IaaS          | Infrastructure as a Service   |
| GCP           | Google Cloud Platform         |
| AWS           | Amazon Web Services           |

# Capítulo 1

## Introdução

### 1.1 Motivação

As organizações procuram, cada vez mais, atingir níveis de profissionalismo elevados e que lhes permitam ser competitivas num mercado global de exigências crescentes. Assim, têm vindo a procurar automatizar o máximo de atividades de natureza produtiva ou a nível administrativo, em especial as que têm um carácter repetitivo e, portanto, exigem um consumo elevado de tempo. Por outro lado, se as atividades se realizam frequentemente as empresas procuram apostar na respetiva simplificação de procedimentos. Com a adoção de mecanismos de automatização dos processos repetitivos e rotineiros ou que manuseiam elevados volumes de dados, é possível a realização das atividades de uma maneira mais eficiente e eficaz, ou seja, com menos erros, menos custos e com menores tempos de execução.

Nem todas as empresas investiram nesta automatização, quer por não terem recursos suficientes para o fazer, por desconhecimento das implicações ou mesmo por não acreditarem que seria uma solução sustentável. Como consequência, algumas organizações ficaram desatualizadas e deixaram de conseguir competir no mercado com as empresas que apostaram na inovação. Investir na automatização e melhoria contínua dos processos é um imperativo e uma necessidade para a sobrevivência num mercado cada vez mais global.

A Management Solutions é uma empresa de consultoria de gestão e de tecnologias que decidiu investir no desenvolvimento de aplicações que melhorassem a produtividade interna das suas equipas e, simultaneamente, criar soluções aplicacionais que tivessem possibilidade de ser colocadas no mercado.

Para isso, são identificados os processos internos que podem ser otimizados e criam-se aplicações que permitam realizar de forma automatizada as respetivas atividades. Ao otimizar o processo a nível interno, a empresa soluciona uma ineficiência interna e quando a solução está suficientemente testada e robusta fica disponível para gerar receitas para o negócio. Pode-se assim recuperar o investimento feito para desenvolver a aplicação internamente.

Esta empresa tem cerca de 2000 profissionais que trabalham em 24 escritórios repartidos pela Europa, América e Ásia, estando presente em mais de 40 países (Management Solutions, Website, 2019).



A atividade de consultoria requer que os consultores façam deslocações constantes aos clientes e posteriormente procedam ao controlo das despesas relacionadas com as viagens, pagamentos de ajudas de custo, alojamento e alimentação. Mensalmente, estas atividades geram milhares de faturas que necessitam de ser registadas, validadas e processadas, envolvendo o controlo através de folhas de despesa. O processamento das folhas de despesa era consumidor de tempo dos consultores e da equipa de suporte, pois envolvia atividades manuais de transcrição de dados das faturas para uma folha de Excel e o posterior carregamento na base de dados da área financeira da empresa. Também a receção de milhares de faturas em papel requeria o consumo de pastas de arquivo e espaço de armazenagem na empresa.

Estes factos foram suficientes para justificar o desenvolvimento de uma aplicação que automatizasse o processo de gestão de despesas dos consultores e, preferencialmente, que o mesmo estivesse a funcionar através de uma aplicação móvel por ser a ferramenta de acesso mais generalizado dentro desta empresa. Com esta solução a empresa reduziria as horas consumidas em trabalhos de processamento de faturas, melhoraria a organização e controlo das despesas nos projetos e diminuiria os custos com material de arquivo, bem como libertaria espaço de arrumação de pastas.

Para a realização deste projeto foram equacionadas diferentes plataformas com serviços que suportam a base do desenvolvimento da aplicação, nomeadamente a Azure, Amazon Web Services e Google Cloud. Foram comparadas as características, vantagens e desvantagens de cada solução na ótica técnica e de adequação às políticas da empresa, para seleção da plataforma a utilizar.

## **1.2 Objetivos**

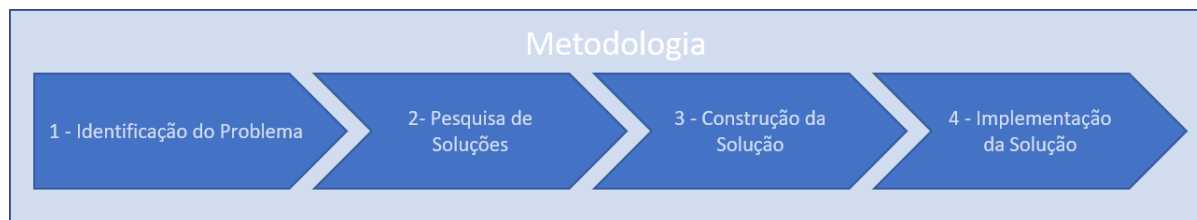
O objetivo geral do presente trabalho foi o de desenvolver uma solução aplicacional que permita a otimização do processo de gestão de despesas da Management Solutions. Esta solução abrange duas abordagens, uma de curto prazo assente na pesquisa por algoritmos específicos e que não é escalável e outra de longo prazo, assente em deep learning e que é escalável dependendo do volume de dados a processar. Para além do objetivo geral existem os seguintes objetivos específicos:

- Desenvolver uma solução numa plataforma da nuvem que a partir de uma foto de uma fatura, tirada por um telemóvel, possa extrair campos específicos, tais como, a designação da empresa que emitiu a fatura, país de emissão, data, valor total, impostos associados, moeda e respetivas subdivisões.
- Armazenar os dados que são considerados relevantes e importantes para gestão e análise de faturas.

- Comprovar o funcionamento da aplicação através de testes unitários.

## 1.3 Metodologia

Para efeitos de realização do presente projeto foi adotada a seguinte metodologia presente na Figura 1.1.



*Figura 1.1: Metodologia*

### 1. Identificação do Problema

- Reunir com os responsáveis da Management Solutions para diagnosticar o problema existente.
- Realizar entrevistas e observar no terreno o funcionamento do processo atual.
- Sistematizar as principais conclusões e validar com os responsáveis da MS.

### 2. Pesquisa de Soluções

- Selecionar entre várias plataformas de Cloud a mais adequada para os objetivos deste projeto.
- Identificar e avaliar dentro da plataforma selecionada a tipologia de serviços requerida para o desenvolvimento da solução.
- Analisar a multiplicidade de faturas que existem em vários escritórios da MS, ou seja, existentes em países diferentes.
- Investigar as melhores práticas na pesquisa e tratamento de campos de faturas.

### 3. Construção da Solução

- Definir os processos aplicacionais a partir dos serviços selecionados.
- Desenvolver a solução para resposta a necessidades de curto prazo, ou seja, uma solução mais generalista não escalável com algoritmos específicos.
- Analisar e extrair dados das faturas que são considerados relevantes para o futuro.
- Construir um modelo de dados que suporte uma solução escalável.
- Desenvolver uma solução interna escalável, suportada por um modelo treinado em Deep Learning.
- Desenvolver uma Interface em android para apresentar os algoritmos desenvolvidos.

#### 4. Implementação da Solução

- Testar a aplicação com múltiplas faturas de vários países.
- Analisar os resultados obtidos.
- Corrigir e afinar a solução desenhada.
- Arrancar em produtivo em Espanha, divulgando o modo de funcionamento da aplicação, esclarecendo dúvidas.
- Alargar o funcionamento da aplicação aos outros escritórios.

## 1.4 Enquadramento Institucional

A MS é uma empresa global que trabalha em diversos países para empresas que são líderes do mercado suportando-se numa equipa de profissionais de excelência. Orientada para o cliente, supera os desafios que lhe são propostos, comprometendo-se a exceder sempre que possível as expectativas dos seus clientes. Trabalha projetos de natureza estratégica em diferentes setores de atividade, tais como setor financeiro, telecomunicações, energia, entre outros. A tipologia de projetos inclui otimização de processos comerciais e de marketing, gestão e controlo de riscos, informação de controlo de gestão e monitorização de resultados e otimização de processos. Todos estes projetos se suportam numa área de tecnologias que alavancam os modelos de negócio das organizações.

Para motivação dos seus quadros a empresa aposta num modelo de gestão de *partnership* em que cada profissional tem a oportunidade de poder atingir o topo da sua carreira e chegar a sócio da empresa num período estimado entre 13 a 16 anos. A carreira está claramente definida e permite que haja promoções internas assentes na meritocracia, no desempenho de cada um e respetivo potencial. A empresa proporciona um bom ambiente de trabalho que permite desenvolver o talento das equipas. Recebe estagiários de diferentes áreas de conhecimento, permitindo que os mesmos trabalhem próximos da gestão de topo da empresa. Os quadros de topo contribuem para o desenvolvimento dos estagiários e desafiam-nos com projetos aliciantes e relevantes para a empresa e para o mercado.

## 1.5 Organização do Documento

O documento encontra-se estruturado de acordo com os seguintes capítulos:

- Capítulo 1 - Introdução: Este capítulo, para além da motivação do projeto, faz a descrição dos objetivos, apresenta sinteticamente a metodologia adotada para a realização do trabalho, descreve o enquadramento institucional da Management Solutions empresa na qual se realizou o estágio académico e finalmente inclui a estrutura deste documento.

- Capítulo 2 – Estado da Arte: Descreve-se os serviços que oferecem três das principais plataformas que trabalham sobre a nuvem, bem como as melhores práticas existentes em termos de técnicas para responder aos objetivos definidos no capítulo 1 deste projeto. Inclui também a sumarização dos pontos fortes e fracos das mesmas plataformas. Faz-se ainda, uma abordagem focada relativa à natural language processing e deep learning.
- Capítulo 3 – Desenvolvimento do Projeto de Tratamento de Faturas: Inclui o detalhe do desenvolvimento deste projeto desde a fase de diagnóstico do problema, identificação de soluções e respetivo desenvolvimento.
- Capítulo 4 – Resultados e Avaliações: Este capítulo inclui a avaliação das duas soluções implementadas e os resultados obtidos identificando as vantagens e algumas debilidades das mesmas.
- Capítulo 5 - Conclusões e Trabalho Futuro: Integra conclusões relativas ao trabalho realizado, bem como identificam-se algumas linhas de intervenção futura que possam contribuir para a melhoria das soluções desenvolvidas no âmbito deste projeto.

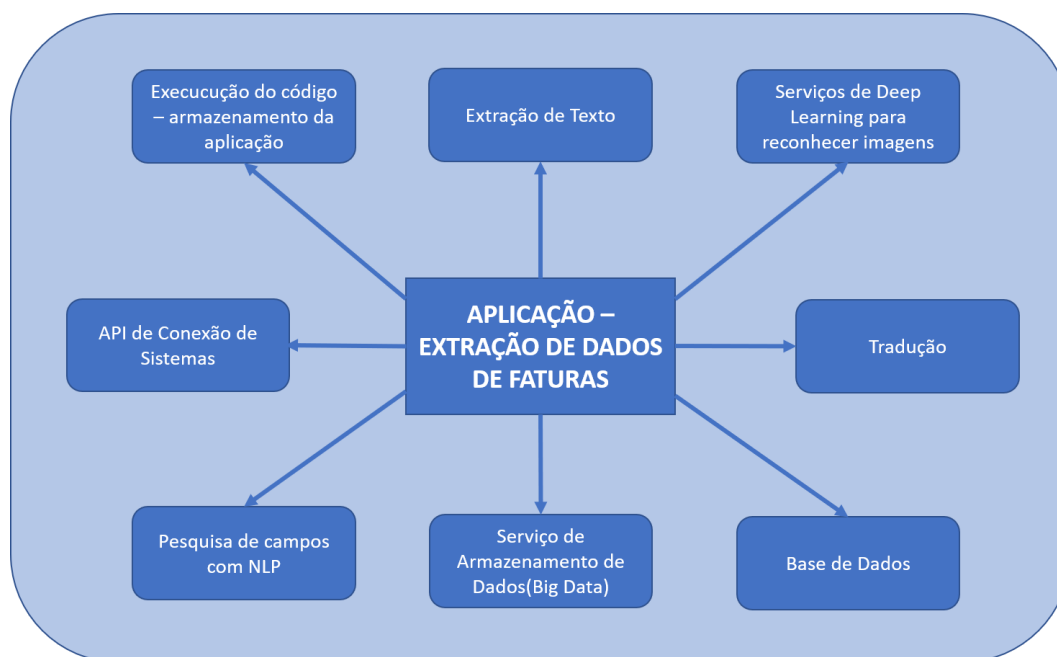
## Capítulo 2

### Estado da Arte

Neste capítulo é realizado um estudo sobre as técnicas que existem no mercado para responder às necessidades deste projeto em termos de tratamento de faturas. Para processar faturas de uma forma automatizada é necessário que, a partir da imagem de uma fatura, se faça a leitura para pesquisar campos específicos que é necessário extrair, eventualmente normalizar os dados extraídos e proceder ao respetivo processamento e armazenamento.

Para este efeito foram comparados os diferentes serviços, técnicas e sempre que aplicável os resultados das soluções de cada um dos seguintes provedores de serviços, que trabalham sobre computação na nuvem – Google, Amazon e Microsoft. Adicionalmente, foram estudadas as aplicações existentes no mercado que processam e extraem dados de faturas, equacionando qual a solução mais adequada à realidade da Management Solutions.

Assim, foi necessário que fosse feito um estudo das características dos vários serviços de tratamento de imagem, a partir de uma aplicação móvel, um serviço de tradução que normalizasse os idiomas constantes das faturas, um OCR que permitisse identificar o texto e respetivas coordenadas. Também se analisaram serviços que permitissem execução de código e armazenamento de grandes volumes de dados e serviços de pesquisa de linguagem de processamento natural para deteção de entidades. A Figura 2.1 sistematiza todos os serviços analisados:



*Figura 2.1: Características necessárias para o desenvolvimento da aplicação.*

## 2.1 Computação na Nuvem

### 2.1.1 Enquadramento

Apesar do termo “computação na nuvem” ser atualmente muito solicitado e considerado fundamental, este conceito já existe desde 1960 e foi apresentado por John McCarthy. Segundo McCarthy “a computação pode um dia ser organizada como um serviço público”. O termo “nuvem” é originário do mundo das telecomunicações, onde estas companhias começaram por oferecer serviços de VPNs com um serviço melhor e custo mais económico. Com recurso às VPNs existe a possibilidade de mudar o tráfego que equilibra a utilização de toda a rede. A computação na nuvem alarga este processo para cobrir a infraestrutura das redes e servidores. (Bairagi, S. I., & Bang, A. O , 2018)

O utilizador tem a possibilidade de escolher o tipo de serviço que necessita sempre que tal seja requerido. Em alternativa à criação de uma rede física, os utilizadores têm a possibilidade de utilizar um serviço que serve como provedor e é administrado pela cloud, ou internet. Os utilizadores têm apenas de pagar os serviços que consumirem. Estes serviços são tratados e sustentados pelas redes que formam a cloud, deste modo o utilizador não terá uma carga de trabalho pesada no seu computador local. (Bairagi, S. I., & Bang, A. O, 2018)

Portanto, a requisição de software e hardware do lado do utilizador é reduzida ou nula. Só é necessário um browser como o Google Chrome para se utilizarem os serviços de computação na nuvem. As principais vantagens da computação na nuvem são:

- Redução do hardware e dos custos de manutenção.
- Acessibilidade em todo o mundo.
- Flexibilidade e processos altamente automatizados em que os clientes não precisam de se preocupar com a atualização do software.
- Qualidade do serviço disponibilizado.

Os serviços de computação na nuvem (Srivastava, P., & Khan, R., 2018) são subdivididos nas seguintes categorias:

1. **SaaS** – A ideia de transportar uma aplicação como um serviço na internet é conhecido como SaaS. Em vez de ser instalado o software num computador pessoal, o utilizador tem acesso ao mesmo via internet. Facilita ao utilizador a gestão de software e hardware complexos. Os utilizadores que põem em prática esta ideia não necessitam de se preocupar com o hardware que gere a aplicação, nem com a sua manutenção e respetiva atualização. A Google Maps ou Google Docs são aplicações desenvolvidas em torno do ideal SaaS.

2. **PaaS** – Esta é uma plataforma de desenvolvimento que é oferecida aos clientes e sobre a qual podem implementar o seu software e código, sendo vulgarmente conhecida como PaaS. O utilizador pode desenvolver as suas aplicações e de seguida executar na infra-estrutura oferecida pelo provedor. O serviço integra uma composição predefinida de sistema operacional e servidor de aplicações que faz a gestão das aplicações. A Ubuntu e Apache são serviços onde o utilizador pode desenvolver as suas aplicações.
3. **IaaS** – A IaaS possibilita que um utilizador aceda a serviços utilizando uma rede ampla, como a Internet. Assim, muitos recursos são fornecidos na forma de armazenamento, rede, hardware e sistema operacional. Um utilizador pode criar uma máquina virtual através do seu login numa das plataformas IaaS.
4. **Nuvem Pública, Privada e Híbrida** - Os serviços de nuvem pública estão disponíveis para qualquer utilizador que paga os serviços que venha a consumir. Os serviços de nuvem privada são oferecidos a utilizadores selecionados e existe uma maior segurança e privacidade fornecida por firewalls e alojamento interno. A nuvem híbrida é uma combinação das nuvens pública e privada e cada nuvem pode ser gerida independentemente, mas a informação e as aplicações podem ser partilhadas entre ambas.

Os benefícios trazidos pela computação na nuvem foram de tal forma significativos que, levaram as maiores companhias tecnológicas como a Google, Amazon e Microsoft a investir montantes avultados para terem um serviço de nuvem competitivo, abrangente nos serviços que proporciona e que responda ao maior número de necessidades dos utilizadores.

Os serviços oferecidos estão em constante atualização face às melhorias e soluções que são descobertas diariamente a nível das tecnologias. Neste trabalho, foi feito um estudo destas três plataformas, primeiro a nível geral e depois a nível específico dos serviços fornecidos, para perceber os benefícios e limitações e poder identificar que plataforma utilizar para desenvolver a aplicação de gestão de faturas.

### **2.1.2 Amazon Web Services**

A Amazon oferece serviços de computação na nuvem a mais de 1 milhão de utilizadores espalhados por 190 países. Um dos seus principais objetivos é disponibilizar aos clientes um serviço com a menor latência possível e maior rendimento. Para concretizá-lo, garantem que os utilizadores mantenham os seus dados e trabalhem a partir de uma região predefinida, sendo esta a mais próxima da sua localização.

A Amazon é a líder no mercado da computação da nuvem há mais de 10 anos, por ser uma organização que aposta na inovação e melhoria constante das suas aplicações. (Varia, J., & Mathew, S., 2014) Os pontos fortes que apresenta são:

- Flexibilidade – A maior diferença entre a AWS e os modelos de tecnologias de informação tradicionais, é a flexibilidade trazida pela nuvem. A mesma permite que as organizações utilizem um modelo de programação, sistemas operativos, bases de dados e arquiteturas nos quais já possuem conhecimentos. Por outro lado, oferece uma combinação de diversas arquiteturas que ajudam as empresas a atender às necessidades específicas dos seus negócios.
- Acessibilidade Económica – O custo de uma aplicação muitas vezes pode ser um problema para algumas organizações porque é necessário um investimento avultado, tanto no software como no hardware. Desta forma, a AWS oferece a possibilidade de as organizações utilizarem hardware, que seria bem mais caro instalando e pagando na sua totalidade, dando a hipótese de apenas pagarem os serviços que utilizarem. Isto permite a não existência de compromissos a curto ou longo prazos.
- Escalabilidade – Numa organização a escalabilidade está normalmente associada ao tipo de infraestrutura e investimento que existe na empresa. Se por alguma razão o fluxo de tráfego duplicasse ou triplicasse teria de existir investimento numa nova infraestrutura. Assim com a AWS existe a possibilidade de as organizações adicionarem, removerem ou modificarem a infraestrutura das suas aplicações, de forma a responder à demanda dos seus clientes e aos requisitos dos respetivos negócios.
- Experiência – Muitas vezes as organizações necessitam de uma plataforma que lhes possa oferecer serviços consolidados e confiáveis. A AWS existe desde 2006, pelo que tem uma vasta experiência no fornecimento de serviços a uma escala global, de forma confiável e segura.
- Segurança - A segurança na nuvem é a prioridade máxima para a Amazon por isso é oferecido aos clientes, um centro de dados e arquitetura na rede que satisfaz os requisitos das organizações mais sensíveis à segurança. Esta segurança é semelhante à segurança de um datacenter local com a diferença de que não é necessária a manutenção de instalações nem de hardware. Neste caso, são utilizadas ferramentas de segurança baseadas em software que monitorizam e protegem o fluxo de informações que entram e saem dos recursos da nuvem. Isto possibilita o desenvolvimento e inovação de uma aplicação num ambiente seguro.
- Diversidade – A Amazon é a plataforma de nuvem mais abrangente do mundo contando com 175 serviços completos de datacenters em todo o mundo.

Os pontos fortes acima referidos evidenciam uma plataforma que dispõe dos mais variados serviços e opções. Não obstante, apresenta também um ponto fraco significativo que se relaciona com o preço (Dutta, P., & Dutta, P, 2019). Muitas vezes é difícil perceber como são definidos os valores dos



vários serviços e torna-se uma tarefa árdua analisar quais são as melhores opções a selecionar para cada serviço requerido.

### 2.1.3 Google Cloud Platform

A Google Cloud Platform tem data centers instalados em 24 regiões e oferece serviços a mais de 200 países. A GCP está a crescer rapidamente, mas é um processo que ainda está em curso. Desta forma, a Google decidiu mudar a sua estratégia e criar parcerias fortes com outras empresas, algo que no passado não ocorria com muita frequência. Uma das parcerias que fez revolucionar os serviços da nuvem foi com a Cisco, que permitiu melhorias significativas nos seus serviços.

O sucesso da GCP existe por conta do desenvolvimento da nuvem nos seus principais pilares escalabilidade e aprendizagem automática. (Bairagi, S. I., & Bang, A. O, 2018) Os pontos fortes da GCP são:

- Segurança – O modelo de segurança da GCP é baseado nas aplicações que são utilizadas pela empresa como o Gmail ou Google Maps. Como já existe a comprovação desta segurança, que é testada há mais de 15 anos, os clientes têm a tranquilidade que lhes é dada por ser utilizado o mesmo modelo.
- Ecológica – Os seus *datacenters* funcionam com metade da energia quando comparados com a concorrência e são alimentados com energia 100% renovável.
- Inovação – A GCP fornece tecnologias inovadoras de Big Data, bem como serviços e bibliotecas de última geração que armazenam dados em nuvem e aprendizagem automática atualizados.
- Performance – A Google oferece uma estrutura confiável, rápida e sólida que dificilmente sofre instabilidades. Os serviços de live-streaming necessitam de uma transmissão de dados eficaz e aplicações como o Spotify e Snapchat confiam na utilização da Google como a sua plataforma.

Apesar da sua robustez esta plataforma possui os seguintes pontos fracos (Dutta, P., & Dutta, P, 2019):

- Funcionalidades e Serviços - Possui menos serviços e funcionalidades do que as outras plataformas, dado que as mesmas ainda se encontram em desenvolvimento.
- Experiência – A GCP é o mais recente dos três fornecedores analisados, o que significa que é o menos maduro em termos de conhecimento e resposta ao mercado.

## 2.1.4 Azure

A Plataforma de nuvem da Microsoft denominada Microsoft Azure tem cerca de 10 anos e teve um início conturbado. Vários críticos consideraram a plataforma muito abaixo da AWS, pelo que houve um investimento para melhoria da plataforma. Com o tempo, os serviços foram-se aperfeiçoando e múltiplas linguagens e frameworks foram apresentadas, incluindo Linux.

Atualmente, a Azure apresenta um portfolio de grande dimensão com muitos serviços da nuvem. Dada a ligação com a Windows vai manter-se numa posição de topo no mercado porque possui recursos quase ilimitados no que se refere a crescimento e mudanças. (Collier, M., & Shahan, R., 2016) Os pontos fortes que a Microsoft Azure apresenta são:

- Disponibilidade – A Microsoft garante 99.5% de tempo de disponibilidade, ou seja, a sua inatividade é menos de 5 horas por ano. Com isto, existe um baixo risco de não disponibilidade de uma aplicação alojada na Azure.
- Suporte – Independentemente do tamanho da aplicação que esteja a ser desenvolvida existe sempre uma equipa de suporte que pode apoiar o projeto a qualquer momento.
- Escalabilidade – Os recursos são facilmente ampliados e reduzidos, sendo fácil atender a qualquer pedido por parte do cliente. Todos os serviços são agrupados num único datacenter corporativo, com isto a latência de rede das aplicações é muito reduzida.
- Recuperação – A Azure apresenta um backup automático que protege os dados do utilizador. Oferece a possibilidade ao cliente de escolher os dados que quer proteger, pelo tempo que quiser com custos reduzidos, pagando apenas o que for utilizado. Também existe a hipótese de restaurar máquinas virtuais inteiras ou pastas.

É reconhecido que apesar da sua robustez a solução da Azure, evidencia os seguintes pontos fracos (Dutta, P., & Dutta, P, 2019):

- Documentação – A Azure apresenta falhas na documentação, sendo referido que muitas vezes é difícil aos clientes entenderem os serviços desenvolvidos.
- Opções de Sistemas Operativos – O suporte da Azure a outros sistemas operativos é muito limitado, abrangendo apenas uma variedade limitada de Linux.

## 2.1.5 Sumarização das plataformas de nuvem

As plataformas analisadas dispõem de muitas vantagens e evidenciam requisitos diversificados e abrangentes, porém com algumas características que as diferenciam.

A Amazon, neste momento, é a plataforma com oferta mais abrangente, mais experiência e com serviços de fácil integração. No entanto, o utilizador depara-se com uma dificuldade quando a mesma apresenta uma secção de preços de difícil compreensão.

Por outro lado, a Azure dá uma boa resposta na integração de soluções com produtos da Microsoft, possuindo também ferramentas de desenvolvimento e teste sólidas, a que se junta um serviço de nuvem híbrida muito completo. Os seus pontos frágeis estão na qualidade da sua documentação que pode, algumas vezes, estar incompleta ou pouco clara e na limitação dos sistemas operativos que oferece ao utilizador.

A GCP oferece descontos e contratos flexíveis nos seus serviços que podem ajudar as empresas que estão no seu início e oferece um serviço direcionado e desenhado para empresas que estão alojadas na nuvem. Porém, sendo a mais recente das soluções não oferece uma abrangência de serviços tão ampla como as outras plataformas, estando ainda na sua fase de crescimento.

Em suma, ao escolher uma plataforma deverá ser feita a análise das necessidades específicas, dos recursos disponíveis e ter-se em atenção as políticas aplicáveis aos sistemas de informação para que a tomada de decisão seja a mais adequada à realidade empresarial.

A Tabela 2.1, abaixo sistematiza os principais pontos fortes e fracos (Paper, W,2015) acima descritos:

| Plataformas           | Pontos Fortes  | Pontos Fracos                             |
|-----------------------|--|---|
| Amazon Web Services   | Flexibilidade<br>Acessibilidade Económica<br>Escalabilidade<br>Experiência<br>Segurança<br>Diversidade | Complexidade                              |
| Google Cloud Platform | Segurança<br>Ecológica<br>Inovação<br>Performance  | Funcionalidades e Serviços<br>Experiência |
| Microsoft Azure       | Disponibilidade<br>Suporte<br>Escalabilidade<br>Recuperação  | Documentação<br>Opções                    |

*Tabela 2.1 Pontos fortes e fracos das várias plataformas de nuvem*

Após a identificação das plataformas principais existentes no mercado foi necessário fazer uma investigação mais detalhada dos serviços das plataformas da nuvem que pudessem ser utilizados para a realização do projeto de extração de dados de faturas. Seguidamente, apresentam-se as técnicas que são necessárias para o desenvolvimento de uma aplicação que faça a extração de dados de faturas, tal como referido na Figura 2.1.

## **2.2 Técnicas Analisadas para a Aplicação**

### **2.2.1 Extração de Texto**

Esta funcionalidade permite a extração dos campos de texto, com recurso a uma imagem de uma fatura.

A Amazon fornece um serviço de OCR que resolve o problema da digitalização de uma fatura que se denomina de Amazon Textract. O Amazon Textract vai além do OCR, porque também identifica o conteúdo dos campos que estão em formulários e informações armazenadas em tabelas. Para além da rapidez e precisão, não requer nem código nem modelos para a sua manutenção, pelo que tem custos baixos de processamento de documentos. (Amazon, Amazon Textract, 2020)

A Google oferece a livreria Cloud Vision que tem um serviço de OCR incluído deteção de texto em imagens, deteção de escrita à mão e deteção de texto em arquivos. Esta solução está otimizada para documentos densos. (Google, Cloud Vision, 2020)

A Azure oferece o serviço Computer Vision que também extrai texto de impresso, suportando sete idiomas e o texto escrito à mão em inglês. (Azure, Computer Vision, 2020)

Para além destes serviços de extração de texto foi identificada a biblioteca Tesseract OCR que funciona com Python e que é *open source*. Permite ler todo o tipo de imagens suportadas pelas bibliotecas de imagens Pillow e Leptonica, incluindo jpeg, png, gif, entre outras, bem como permite a impressão do texto que reconheceu.

### **2.2.2 Tradução**

A nível da tradução os serviços requeridos devem permitir interpretar texto multilíngue e normalizar num único idioma, neste caso o inglês, aspetos chave para a criação de um dataset eficaz que defina o tipo de fatura, diminuindo o número de erros.

A Amazon dispõe do Amazon Translate, que é um serviço de tradução automático neural que faz tradução rápida, de alta qualidade e a preços acessíveis. Utiliza modelos de aprendizagem profunda para oferecer uma tradução mais precisa e natural do que os tradicionais algoritmos estatísticos e de tradução baseados em regras. Promove a melhoria contínua do sistema de tradução, é de fácil integração com outras aplicações e é escalável. (Amazon, Amazon Translate,2020)

A Google possui um serviço de traduções dinâmicas entre idiomas denominado Translation e que se suporta em modelos de ML personalizados ou pré-treinados da Google. Permite treinar modelos entre idiomas bastando fazer o upload de pares de idiomas traduzidos e o AutoML Translation cria um modelo personalizado. A API Translation converte textos de imediato em mais de 100 idiomas. Permite personalizar as traduções para adequar termos e frases específicas em função do contexto em que se inserem. Adicionalmente, deteta idiomas com precisão, é de fácil integração e oferece preços acessíveis. (Google, Cloud Translation,2020)

A Azure oferece o serviço Translator dentro dos Cognitive Services, que permite fazer traduções para 60 idiomas, suportando-se em ML. Permite fazer modelos de tradução com terminologia específica adaptável a necessidades próprias dos clientes. Para os clientes da Microsoft é muito vantajoso pois existe uma integração entre o Translator e produtos Microsoft. (Azure, Translator,2020)

Foi ainda analisada a biblioteca Translate desenvolvida em Python,, que utiliza a API da Microsoft com o serviço Translator, permitindo fazer traduções de uma forma simples e podendo ser utilizada diretamente na linha de comandos integrando facilmente com outros produtos da Microsoft.

### **2.2.3 Execução do código – Armazenamento da Aplicação**

O código desenvolvido para o tratamento de faturas necessita de ser armazenado e executado num servidor pelo que se procedeu à identificação das possibilidades oferecidas pelas diferentes plataformas. O processo de gestão de servidores, às vezes, pode ser exaustivo e há muitos processos que precisam de ser executados para uma aplicação funcionar. Foram consideradas as soluções disponibilizadas pelas três plataformas principais.

A Amazon, apresenta o Amazon Lambda, um serviço que permite executar código sem provisionar ou gerir servidores, ou seja, executar código para qualquer tipo de aplicação ou serviço backend sem necessidade de serviços de administração. Após a identificação dos serviços necessários a utilizar, configura-se o código que permite o funcionamento automático ou sempre que houver uma solicitação do respetivo código. Além de não haver necessidade de administração de servidores, é escalável, o pagamento está associado à frequência e ao tempo de utilização e o seu desempenho é reconhecido como sendo consistente. As Lambdas podem ser executadas por inúmeras linguagens de programação como: Python, Java ou Node JS (Amazon, Amazon Lambda, 2020). A Amazon também disponibiliza o Amazon EC2 que é um serviço que permite que os utilizadores aluguem máquinas virtuais para as suas aplicações. O acesso é fácil, dado que tem interfaces web que permitem aos utilizadores processarem de acordo com as suas necessidades mantendo o controlo completo do processo e a segurança, bem como lhes permite selecionar o tipo de distribuição desejada como Windows ou Linux. (Amazon, Amazon EC2, 2020)

A Google tem uma plataforma de computação que permite a execução de código sem servidor designada Cloud Functions. É escalável e o custo depende dos recursos de computação utilizados. No âmbito desta plataforma podem ser utilizadas várias linguagens de programação, existindo uma monitorização, registo e rastreio dos processos executados, bem como processos de segurança integrada com limites bem definidos. (Google, Cloud Functions, 2020)

Para além desta solução existe a Google Kubernetes Engine que é um sistema que utiliza contentores, permitindo uma melhor delimitação de responsabilidades pois é possível isolar os processos de cada ferramenta. A infraestrutura fica mais eficiente e flexível potenciando ganhos em escala e performance. (Google, Google Kubernetes, 2020)

A Azure apresenta o serviço Azure Functions que é uma solução escalável e flexível dependente do volume de dados da aplicação. Oferece um modelo de programação integrado com outros serviços, assente em ligações e acionadores que respondem a eventos. Suporta muitas linguagens de programação e os utilizadores podem com facilidade ajustar-se para dar resposta aos requisitos da aplicação. (Azure, Azure Functions, 2020) A Azure também oferece o serviço Virtual Machines que possibilita ao utilizador executar as suas aplicações num meio de computação de alto desempenho, escolhendo entre as distribuições Windows Server ou Linux. Este modelo permite a otimização da infraestrutura e redução de custos. (Azure, Azure Virtual Machines, 2020)

Para além destes provedores existem outros que também disponibilizam serviços similares como a Kamatera, Vultr ou BlueHost, mas não foram alvo de investigação.

## 2.2.4 Pesquisa de campos com NLP

Um dos problemas identificados para processar as faturas, refere-se ao elevado volume de dados, à informação estar em vários idiomas, com estruturas diferentes, pelo que a utilização de ferramentas assentes no NLP se tornou essencial para o sucesso deste projeto.

A Amazon apresenta o serviço Amazon Comprehend que utiliza modelos de aprendizagem profunda e de aprendizagem automática para identificar características específicas dos dados com que trabalha. O Amazon Comprehend é um serviço orientado à análise e tratamento de texto, cujo objetivo é extrair e processar o máximo de informação. Utiliza a aprendizagem automática para encontrar *insights* e relacionamentos em texto, sendo simples e estando disponível para qualquer pessoa, não sendo necessário conhecer machine learning. Com esta aplicação é possível organizar os documentos por tópicos, treinar os modelos com os seus dados e tem suporte para textos gerais ou específicos. (Amazon, Amazon Comprehend, 2020) Apresenta ainda o serviço Amazon Sagemaker que permite que um utilizador possa criar, treinar e implantar modelos de Machine Learning próprios, de modo fácil e com alta qualidade. Suporta-se e disponibiliza um conjunto único de ferramentas que integram todos os componentes usados para o Machine Learning. (Amazon, Amazon Sagemaker, 2020)

A Google oferece o serviço Natural Language que permite aos desenvolvedores trabalhar com recursos de processamento de linguagem natural, incluindo análise de sentimento, entidades, classificação de conteúdo e análise sintática, procedendo ao armazenamento dos documentos na cloud storage. (Google, Natural Language, 2020)

A Azure disponibiliza o serviço Text Analytics que deteta informações em texto não estruturado com o processamento de NLP, podendo ser utilizado por pessoas que não possuam conhecimento de Machine Learning. Suporta-se na identificação de expressões chave e entidades como pessoas, locais e organizações para poder compreender os temas abordados e mesmo tendências. (Azure, Text Analytics, 2020)

Foram ainda analisadas várias bibliotecas *open source* desenvolvidas em Python, que é a linguagem mais popular para trabalhar no campo de NLP:

- Spacy – Desenhada para ajudar os utilizadores a interpretar temas que surgem numa análise de tópicos. Permite visualizar de uma forma simples cada um dos temas incluídos no texto.
- TextBlob – Simplifica o processamento de texto proporcionando uma interface intuitiva. Possui uma curva de aprendizagem suave, com muitas funcionalidades úteis e inovadoras.

- NLTK – É a biblioteca mais utilizada para NLP, proporcionando classes fáceis de utilizar com mais de 50 corpus e recursos léxicos, em conjunto com bibliotecas de processamento de texto para classificação, tokenização, análise e raciocínio semântico.
- Stanford CoreNLP – Desenvolvida pela Universidade de Stanford, constituída por muitas técnicas tradicionais de NLP.
- Textacy – É uma biblioteca muito robusta desenhada sobre Spacy para facilitar várias tarefas relacionadas com NLP.
- Gensim – Extrai automaticamente os temas semânticos dos documentos de forma eficiente e simples.
- Scikit-learn – Integra facilmente com outras bibliotecas Python como a matplotlib e plotly para impressão de gráficos, numpy para vectorização de matrizes, pandas para construção de dataframes, entre outros.

### **2.2.5 API para Conexão de Sistemas**

As API's são essenciais na integração de sistemas pois protegem e dão segurança dos dados tratados, automatizam processos e facilitam a troca de informação entre sistemas com diferentes linguagens de programação. As API's funcionam como a porta de entrada para aplicações de acesso a dados ou funcionalidades de serviços backend. Para o desenvolvimento da aplicação é necessária uma API que permita conectar rotinas de um modo simples e enviar respostas para o utilizador.

A Amazon oferece o serviço Amazon API Gateway que é automatizado e facilita aos desenvolvedores a criação, publicação, manutenção, monitorização e segurança de API's em qualquer escala. A Amazon API Gateway gere as tarefas envolvidas na aceitação e processamento de um volume elevado de chamadas de API's simultâneas, incluindo gestão de tráfego, suporte CORS, controlo de autorização e acesso, monitorização e gestão de versões da API. Este serviço permite o desempenho em qualquer escala, com economia de custos por ser escalável e com controlos de segurança flexíveis. (Amazon, Amazon API Gateway, 2020)

A Google fornece o serviço Cloud Endpoints que assenta numa arquitetura distribuída e de proxy com base em NGINX, proporcionando um elevado desempenho, consistência e escalabilidade. Oferece também as ferramentas para cada etapa do desenvolvimento apoiando através do Cloud Logging, Cloud Monitoring e Cloud Trace. (Google, Cloud Endpoints, 2020)

A Azure disponibiliza o serviço API Management que permite uniformizar o trabalho desenvolvido em ambientes híbridos e com várias clouds numa única localização, na qual se faz a gestão



de todas as API's. Permite criar portais personalizáveis dos programadores e fazer partilhas das API's com equipas internas, parceiros externos e clientes. (Azure, API Management, 2020)

Para além destes provedores foram investigadas outras opções que assentam em Python das quais o Flask e FastAPI. Ambas são rápidas, fáceis de desenvolver, com poucos erros sendo ainda muito intuitivas e robustas.

## **2.2.6 Serviço de Armazenamento de Dados (Big Data)**

A informação contida nas faturas é sensível e crítica para a organização quer a nível interno, quer para prestação de contas a entidades externas como os acionistas ou o fisco. Assim, foi necessário salvaguardar o seu armazenamento numa plataforma sem recurso a papel, permitindo um ganho para a empresa, em espaço e custos. A criticidade desta informação exigiu que a solução a adotar pudesse armazenar elevados volumes de dados por longos períodos e com o máximo de informação contida nas faturas.

A Amazon oferece o Amazon Simple Storage Service (Amazon S3) que é um serviço de armazenamento de objetos com escalabilidade, disponibilidade de dados, segurança e desempenho. A Amazon S3 fornece recursos de gestão fáceis de utilizar para que se possa organizar dados e configurar controlos de acesso ajustados aos requisitos específicos de negócio. Com os pontos de acesso S3, podem-se criar centenas de pontos de acesso por bucket, representando uma nova forma de facilitar o acesso a dados partilhados. Inclui ainda, serviços de query-in-place para análise, bem como um robusto sistema de segurança com capacidades de auditoria e conformidade. (Amazon, Amazon S3, 2020)

A Google dispõe do serviço Google Cloud Storage que é uma opção de armazenamento flexível e durável, podendo ser utilizada a partir de qualquer localização. Este serviço combina o desempenho e a escalabilidade da nuvem com recursos avançados de segurança e partilha. (Google, Cloud Storage, 2020)

A Azure disponibiliza o serviço Azure Storage que é uma solução para vários cenários de armazenamento de informação. Inclui serviços escaláveis desenhados para poder dar resposta a volumes significativos de dados e às necessidades de desempenho das aplicações atuais. Tem sistemas de segurança encriptados oferecendo controlo de acesso aos dados e prevenindo situações de falhas no Hardware por possuir sistemas de redundância. (Azure, Azure Storage, 2020)

Para além destes provedores existem outros que também disponibilizam serviços similares como a IBM, Dell EMC ou HPE, mas não foram alvo de investigação.

### **2.2.7 Base de Dados – NoSQL**

Os dados extraídos das faturas necessitam de ser guardados numa base de dados para poderem ser consultados e analisados em qualquer momento. Pretende-se que a base de dados assegure consistência nas informações armazenadas, permita a consulta acessível aos dados e exista uma tolerância no tipo de informações armazenadas.

A Amazon possui um serviço de Base de Dados denominado Amazon DynamoDB que suporta estruturas de dados de valor chave e documentos, ou seja, é uma base de dados NoSQL. Possibilita uma performance de excelência com tempos de resposta abaixo de 10 milissegundos em qualquer escala. Expande e reduz tabelas automaticamente ajustando as capacidades para manter níveis de resposta. A disponibilidade e tolerância a falhas já incorporadas nesta solução diminuem as preocupações dos utilizadores em terem que incorporar estes recursos nas suas aplicações. (Amazon, Amazon Dynamo DB, 2020)

A Google dispõe do serviço de base de dados Google Datastore que é altamente escalável para as suas aplicações. Controla a fragmentação e replicação de dados para assegurar que a base de dados está disponível, é rápida e tem durabilidade. Fornece recursos, como consultas a SQL, índices e transações ACID. É uma base de dados simples, a linguagem de consulta é fácil de utilizar. (Google, Google Datastore, 2020)

A Azure tem o serviço de base de dados multimodelo denominado Cosmos DB, facilmente escalável com a disponibilização de resposta aos utilizadores em milissegundos. Existe a possibilidade de utilizar API's de bases de dados como a SQL, a MongoDB, a Cassandra, entre outras, para aceder rapidamente. Diferencia-se positivamente pelos acordos abrangentes de nível de serviço para produção, latência, disponibilidade e garantias de consistência. (Azure, Cosmos DB, 2020)

Para além da análise dos provedores principais da nuvem foram analisadas outras alternativas de bases de dados. A MongoDB é uma base de dados NoSQL, orientada a documentos, direcionada para Big Data, facilmente escalável. Suporta queries dinâmicas para consulta, similares ao SQL.

### 2.2.8 Serviços de Deep Learning para reconhecer imagens

Uma das necessidades da aplicação era ter uma solução para longo prazo, ou seja, que fosse continuamente aperfeiçoando a sua capacidade de análise e resposta. O modelo desenvolvido no início não dispunha de dados, porém no decorrer do processo foram sendo introduzidas mais faturas no dataset. Desta forma, foi possível desenvolver uma solução que analisava estas faturas, cada vez mais diversificadas, e ia aprendendo a ser cada vez melhor no reconhecimento de campos. Para este caso específico entendeu-se como mais adequado uma solução que se focasse no reconhecimento de campos de faturas nas imagens, dividindo em *boarding boxes*, ou seja isto implicava o desenvolvimento de uma rede convolucional. Para melhor compreensão deste tema, antes de analisarmos as soluções das várias plataformas será feita uma abordagem geral ao Deep Learning.

A Amazon possui o serviço Amazon Rekognition que facilita a adição de análises de imagem e vídeo nas suas aplicações, recorrendo a uma tecnologia de qualidade reconhecida e escalável de aprendizagem profunda, que não requer conhecimentos de machine learning. Com Rekognition, é possível identificar objetos, pessoas, texto, cenas e atividades em imagens e vídeos. Adicionalmente fornece recursos de análise e pesquisa facial muito precisos que permitem detectar, analisar e comparar rostos. (Amazon, Amazon Rekognition, 2020) Com o Amazon Sagemaker é possível criar os modelos à medida das necessidades de cada utilizador. (Amazon, Amazon Sagemaker, 2020)

A Google dispõe do serviço Google Vision que permite a funcionalidade de deteção de objetos em imagens, incluindo a localização e classificação de cada objeto na imagem. (Google, Cloud Vision, 2020)

A Azure oferece o serviço Computer Vision que permite o processamento de dados visuais para identificar conteúdo, desde objetos a conceitos, não sendo necessário conhecimentos prévios em Machine Learning para a sua utilização. (Azure, Computer Vision, 2020)

Foram também analisadas bibliotecas desenvolvidas em Python, descrevendo-se as mais relevantes para este projeto:

- TensorFlow – É uma framework desenvolvida pela Google. Está em código aberto para computação numérica usando grafos de fluxo de dados.
- PyTorch – É uma framework de Deep Learning e tem o suporte do Facebook.
- Theano – Permite definir, otimizar e avaliar expressões matemáticas que invocam tensores de maneira eficiente.

- CNTK – São ferramentas, desenvolvidas pela Microsoft, fáceis de utilizar, de código aberto que treinam algoritmos de Deep Learning para aprender, como acontece com o cérebro humano.
- Keras – É uma biblioteca de alto nível, fácil de utilizar. Está escrita e é mantida pela Google Brain. Permite aos utilizadores selecionarem os seus modelos que se constroem, executando-os num grafo simbólico de Theano, TensorFlow ou CNTK.
- MXNET – É uma biblioteca flexível e eficiente para desenvolver modelos de Deep Learning com suporte para vários idiomas.

### 2.2.9 Sistematização das Técnicas Analisadas

Neste capítulo apresentaram-se todas as ferramentas que foram alvo de pesquisa para permitir identificar as que são mais adequadas para suporte ao desenvolvimento de uma solução de tratamento de faturas para a Management Solutions, que se sistematizam na Tabela 2.2:

| Características                              | AWS                                    | GC                                   | Azure                                     | Outros  |
|--|--|--------------------------------------|---|---|
| Extração de texto                            | Amazon Textract                        | Cloud Vision                         | Computer Vision                           | Tesseract OCR   |
| Serviços de Deep Learning                    | Amazon Rekognition<br>Amazon SageMaker | Google Vision                        | Computer Vision                           | TensorFlow<br>PyTorch<br>Theano<br>CNTK<br>Keras<br>MXNET                             |
| Pesquisa de campos com NLP                   | Amazon Comprehend<br>Amazon SageMaker  | Natural Language                     | Text Analytics                            | Spacy<br>TextBlob<br>NLTK<br>Stanford<br>CoreNLP<br>Textacy<br>Gensim<br>Scikit-learn |
| Serviço de Armazenamento de Dados (Big Data) | Amazon S3                              | Google Cloud Storage                 | Azure Storage                             | IBM<br>Dell EMC<br>HPE  |
| API  | Amazon API Gateway                     | Cloud Endpoints                      | API Management                            | FastAPI<br>Flask  |
| Serviço de Base de Dados                     | Amazon DynamoDB                        | Google Datastore                     | Cosmos DB                                 | MongoDB   |
| Tradução                                     | Amazon Translate                       | Google Translation                   | Azure Translator                          | Translate   |
| Execução de Código                           | Amazon Lambda<br>Amazon EC2            | Cloud Functions<br>Google Kubernetes | Azure Functions<br>Azure Virtual Machines | Kamatera<br>Vultr<br>BlueHost   |

*Tabela 2.2 Sistematização das funcionalidades necessárias para a aplicação apresentadas pelos principais serviços de nuvem existentes no mercado*

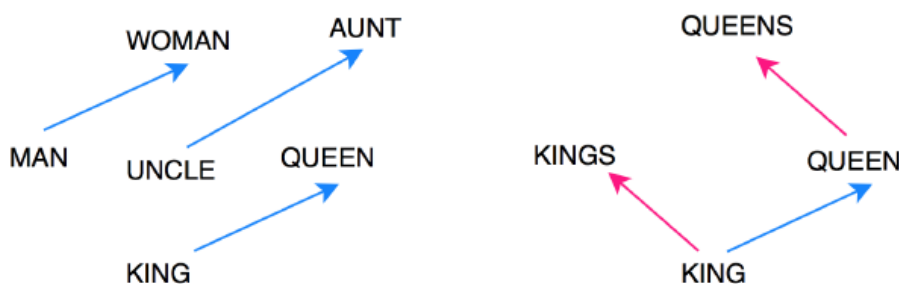
## 2.3 Natural Language Processing

O processamento de linguagem natural é um tópico que se encontra na interseção de várias ciências, tais como, ciência da computação, inteligência artificial e psicologia cognitiva. A ideia

fundamental subjacente ao NLP é fazer com que as máquinas compreendam e leiam os mesmos idiomas que o ser humano. Os algoritmos de NLP baseiam-se na aprendizagem automática ou profunda. Em vez, de se codificarem grandes conjuntos de dados, de forma manual, o NLP pode recorrer a técnicas de aprendizagem automática que ensinam estas regras às máquinas a partir da análise de um conjunto de exemplos construindo estatísticas inferenciais. Em geral, quantos mais dados analisados, mais rigoroso é o modelo e a título de exemplo os algoritmos podem ser utilizados:

- Chatbots e Resumo de Texto – Recorre a técnicas de NLP para extrair as ideias mais importantes e centrais, excluindo informações irrelevantes.
- Geração automática de palavras – Realiza com o NLP a análise do conteúdo associando palavras chave a parágrafos de texto.
- Reconhecimento de entidades – Identifica com o NLP diferentes entidades do texto como uma pessoa, lugar ou organização.
- Análise de sentimentos – Utiliza o NLP para identificar o sentimento expresso num texto, desde negativo, neutral ou positivo.

Para efeitos de associação dos campos na extração de faturas foi identificada a possibilidade de recorrer ao Word Embedding que é uma componente do NLP, cujo objetivo é encontrar similaridades entre palavras. Para a realização da aplicação de tratamento de faturas era essencial conseguir estabelecer relações entre entidades e palavras. (Demeester, T., Rocktäschel, T., & Riedel, S, 2016) Quando o Word Embedding analisa várias palavras mapeia cada palavra, medindo a similaridade entre as mesmas. Se duas palavras forem semelhantes no seu significado, o seu Word Embedding vai ser equivalente. Na Figura 2.2 é apresentado um exemplo que se detalha seguidamente:



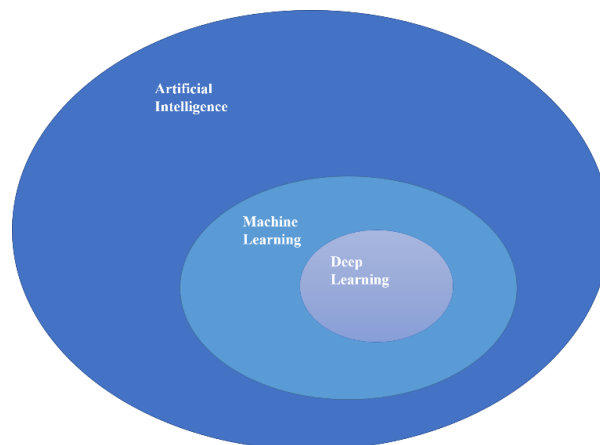
*Figura 2.2: Representação de Word Embedding no espaço*

A relação male/female continua na mesma, relativamente a estes pares, e o word embedding trata consistentemente a relação de género. O mesmo conceito existe em relações sintáticas. Em NLP, existe a possibilidade de designar vetores aleatórios às palavras, construindo o word embedding e com

base em importância e modelos de treino, aperfeiçoar a detecção e ligações entre as palavras. Existem múltiplos modelos treinados e desenvolvidos, dos quais se destacam o Word2Vec e Glove.

## 2.4 Deep Learning

De acordo com (Chollet,2018) apresenta-se na figura 2.3 uma representação onde se pode situar a aprendizagem profunda, sendo esta uma subárea de Machine Learning. Chollet, descreve no seu livro uma série de técnicas avançadas em aprendizagem profunda utilizadas na indústria e estudos académicos, suportados na linguagem de programação Python.



*Figura 2.3: Representação das técnicas: Inteligência Artificial, Aprendizagem Automática e Aprendizagem Profunda. (Chollet,2018)*

Nos últimos anos, com o avanço na tecnologia e na computação, a aprendizagem profunda tem vindo a tornar-se cada vez mais sofisticada e utilizada em aplicações complexas. A disponibilidade de maiores datasets para treino e as melhorias no hardware, como CPU/GPUs mais rápidas, permitiram um crescimento da aprendizagem profunda. (Chollet,2018) A aprendizagem profunda é uma técnica em inteligência artificial, que se baseia num conjunto de algoritmos que permitem construir abstrações de um grande número de dados utilizando um grafo profundo com várias camadas de processamento, compostas por várias transformações lineares e não lineares.

Para resolver um problema de aprendizagem profunda, faz-se a extração de dados, analisa-se o tipo de dados com que se está a trabalhar, cria-se e ajusta-se o dataset e estabelece-se um objetivo realista que permite dar a resposta ao referido problema. Numa situação em que se pretenda reconhecer uma determinada pessoa, se se fizer uma classificação de imagens, há que ter em conta quais as características que devem ser analisadas, como as múltiplas posições do corpo ou a forma do corpo. A complexidade desta interpretação requer um entendimento sofisticado dos dados que vão ser trabalhados. Se se quiser, desenvolver um modelo sobre classificação de imagens é necessário explorar

os dados e dividi-los em inúmeras categorias. Dado estar-se perante fatores de variação, que podem afetar os dados, a extração de categorias de alto nível e categorias abstratas pode ser complicada. Os resultados podem estar incompletos e validá-los é um processo que requer bastante tempo.

Os modelos de aprendizagem profunda lidam com estes problemas desenvolvendo conceitos mais avançados, baseando-se em modelos simples. A visualização da criação destes conceitos é apresentada como um grafo profundo com várias camadas, ou seja, uma rede neural. (Chollet,2018)

### 2.4.1 Rede Neural Profunda

O tipo mais comum de rede neural artificial apresentado em muitos estudos é a rede neural feed-forward. A primeira camada numa rede feed-forward é o seu input, em seguida acrescenta-se uma ou mais camadas escondidas e por fim há uma camada do output. Se a rede for constituída por várias camadas escondidas, considera-se que é uma rede neural profunda, conforme pode ser observado na Figura 2.4.

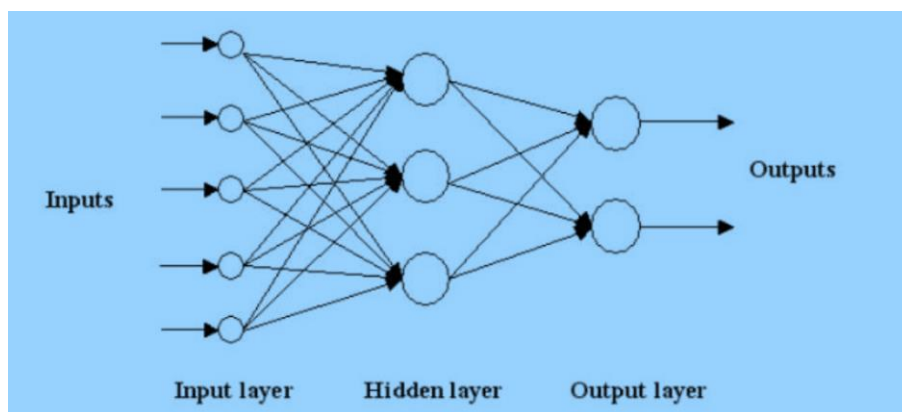


Figura 2.4: Rede neural feed-forward multi-camada. (Chollet,2018)

Cada camada nesta rede é constituída por nós cuja função é realizar uma operação, o que envolve inputs com coeficientes para aumentar ou diminuir a importância desse nó. Suponha-se que se quer definir as características que permitam num conjunto de imagens identificar se a imagem corresponde ou não a um carro. Terá que se manipular o peso de cada input, conseguindo deste modo diminuir o erro da classificação. Na rede neural *feed-forward* o algoritmo *backpropagation* é muito utilizado. Este algoritmo possui um meio de atualização dos seus pesos, através de um vetor gradiente no qual cada elemento é definido por uma medida de erro em relação a um parâmetro. Os erros são definidos pela diferença dos outputs da rede atual pelos outputs desejados, o que implica a existência de outputs disponíveis para o treino. Por essa razão, o backpropagation é um algoritmo de aprendizagem supervisionada.

## 2.4.2 Rede Neural Convolucional

As CNN caracterizam-se pelos seus inputs iniciais serem compostos por imagens. Isto determina que a arquitetura a ser configurada seja preparada para lidar com este tipo de dados. Uma das principais características deste tipo de rede neural é a sua disposição em três dimensões espaciais: altura, largura e profundidade. Na prática, isto significaria que existiria um input de dimensões  $64 \times 64 \times 3$  (altura, largura e profundidade), levando a um output composto por uma dimensionalidade de  $1 \times 1 \times n$  (onde  $n$  representa o número possível de classes). Isto ocorre porque se diminui a dimensionalidade de entrada num volume menor de notas de classe guardadas através da dimensão da profundidade. As CNN são compostas por três tipos de camadas. Existem as camadas convolutivas, camadas de pooling e camadas totalmente conectadas. Quando estas camadas são conectadas, uma arquitetura CNN é formada. Uma arquitetura CNN simplificada para a classificação MNIST é ilustrada na Figura 2.5. (Chollet,2018)

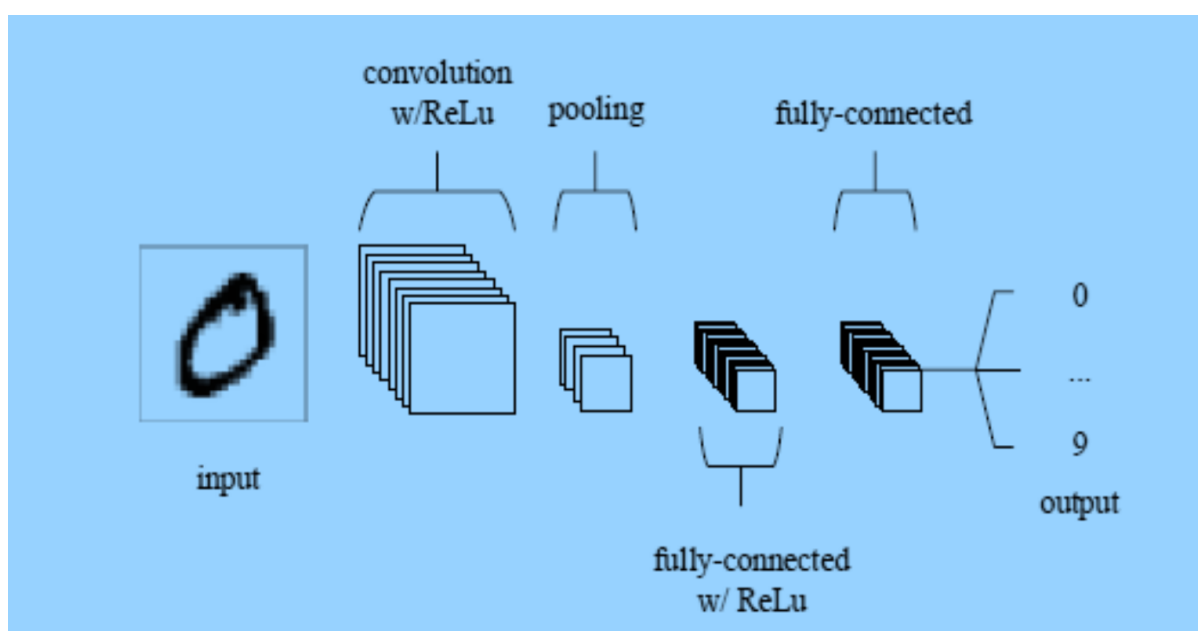


Figura 2.5: Uma arquitetura CNN comprimida com cinco camadas. (Chollet,2018)

As funcionalidades do exemplo apresentado anteriormente na Figura 2.5 podem ser divididas em quatro grupos:

1. A camada de input mantém os valores dos pixels da imagem.
2. A camada convolucional determina o output dos neurónios, os quais estão conectados às regiões locais da entrada, através do cálculo do produto escalar entre os seus pesos e a região conectada ao volume de entrada.



3. A camada de pooling realiza o processo de downsampling, ao longo da dimensão espacial do input recebido, reduzindo o número de parâmetros dentro dessa ativação.
4. As camadas totalmente conectadas executam as mesmas funcionalidades encontradas e produzem pontuações da classe, a partir das ativações, para serem utilizadas na classificação, por tentativa e melhoria contínua.

As CNN são uma classe de rede neural artificial profunda, que tem sido utilizada para produzir um desempenho preciso em tarefas de reconhecimento, classificação e detecção de objetos em imagens. Os testes que têm vindo a ser desenvolvidos com este tipo de redes apresentam resultados muito positivos. Foram desenvolvidos nos últimos anos modelos que foram treinados com milhares de imagens e serviram de prova de conceito para validar a eficiência das CNN's no tratamento de imagem. Por exemplo, a Faster-RCNN é um modelo que foi treinado com milhares de imagens e a sua principal vantagem é a sua precisão e computação rápida.

## **Capítulo 3**

### **Desenvolvimento do Projeto de Tratamento de Faturas**

Este projeto surge no âmbito da colaboração da Faculdade de Ciências e da empresa Management Solutions, que proporcionou a realização de um estágio académico a um aluno desta Faculdade. Foi estipulado que no período de 9 meses, de setembro de 2019 a junho de 2020, fosse desenvolvido um trabalho de relevância para a empresa e que pudesse ser a base para a realização da tese de mestrado.

#### **3.1 Diagnóstico**

A empresa Management Solutions considerou crítica a necessidade de desenvolver uma aplicação que extraísse automaticamente dados de faturas, isto porque mensalmente, a empresa tem que gerir muitas faturas referentes a despesas dos consultores decorrentes da realização de projetos.

O tratamento das faturas implica muitas atividades manuais por parte da equipa de backoffice e dos consultores. No âmbito da otimização de processos internos, constatou-se ser possível realizar um projeto que minimizasse as tarefas administrativas e consequentemente automatizasse algumas atividades rotineiras e repetitivas.

O diagnóstico da situação atual foi feito através de reuniões com responsáveis da MS da área de Inovação e Desenvolvimento, permitindo clarificar os objetivos da solução a desenvolver e o respetivo âmbito.

O objetivo definido foi o desenvolvimento de uma aplicação que tratasse faturas de despesas de projeto dos colaboradores para os vários escritórios da MS. O projeto deveria compreender duas soluções, uma para dar resposta imediata no ano de 2020 e outra para solucionar necessidades de longo prazo que fosse mais eficiente que a anterior.

Após a clarificação dos objetivos do projeto foi analisada a tipologia de faturas para se compreender as necessidades reais e como se poderia extrair a informação. Esse trabalho foi feito com vários colaboradores da MS que apresentaram as faturas mais usuais.

Decorrente desta análise sistematiza-se na Tabela 3.1 os campos específicos para extração que teriam que ser implementados para as duas soluções a desenvolver.

| Campo              | Tipo             | Descrição  | Mostrar na Aplicação |
|--------------------|------------------|--|----------------------|
| País               | Letras           | País da Fatura   | Sim                  |
| Moeda              | Letras           | Moeda da Fatura  | Sim                  |
| Convertor de Moeda | Letras           | Converte em Euros se a moeda não for essa.   | Sim                  |
| Número Fiscal      | Letras e Números | 9 caracteres (1 letra + 8 números + 1 letra ou número)   | Sim                  |
| Data da Fatura     | Data             | Data da emissão da fatura  | Sim                  |
| Nome da empresa    | Letras           | Nome da empresa da fatura analisada  | Sim                  |
| Tipo de Fatura     | Letras           | Tipo de gasto da fatura  | Sim                  |
| Valor total        | Números          | Valor total da fatura  | Sim                  |
| Tipo Impositivo %  | Números          | Apresentar os IVAs da fatura:<br>Tipo de IVA em Espanha:<br>Tipo geral: 21%<br>Tipo reduzido: 10%<br>Tipo super-reduzido: 4% | Sim                  |
| Base Tributável    | Números          | Base Tributável da fatura  | Sim                  |

*Tabela 3.1 Campos específicos para extração*

### 3.1.1 Tipologia de Faturas

As soluções foram equacionadas para abranger a maior amostra possível de faturas. As Figuras 3.1, 3.2 e 3.3 representam faturas reais e verifica-se que podem existir muitas diferenças nos campos, formatos e propriedades. Assim, é difícil desenvolver algoritmos que possam dar resposta em simultâneo a todas as tipologias de faturas. Na primeira solução a abordagem seguida foi de desenvolver algoritmos que permitissem tratar o maior número de faturas possível, sabendo à partida que algumas poderiam não ser processadas de modo eficaz pelo algoritmo. Já na solução de longo prazo, a própria aplicação iria desenvolver mecanismos de melhoria contínua que minimizariam os erros.

A Figura 3.1 representa uma fatura ideal porque apresenta todos os campos definidos como necessários no âmbito do projeto e existe lógica na formatação da fatura.

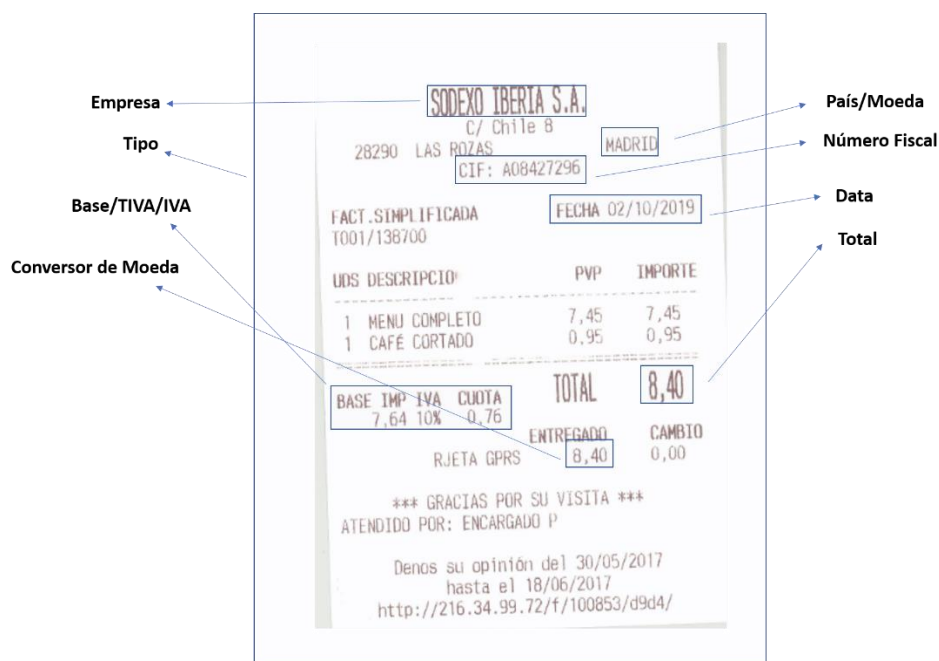


Figura 3.1: Exemplo da primeira fatura com todos os campos

A Figura 3.2, apresenta uma fatura com praticamente todos os campos, mas esta estrutura poderá não permitir a identificação de alguns dos campos. Neste caso, a procura com base num algoritmo, não permitiria encontrar o país porque não existe uma referência de identificação.

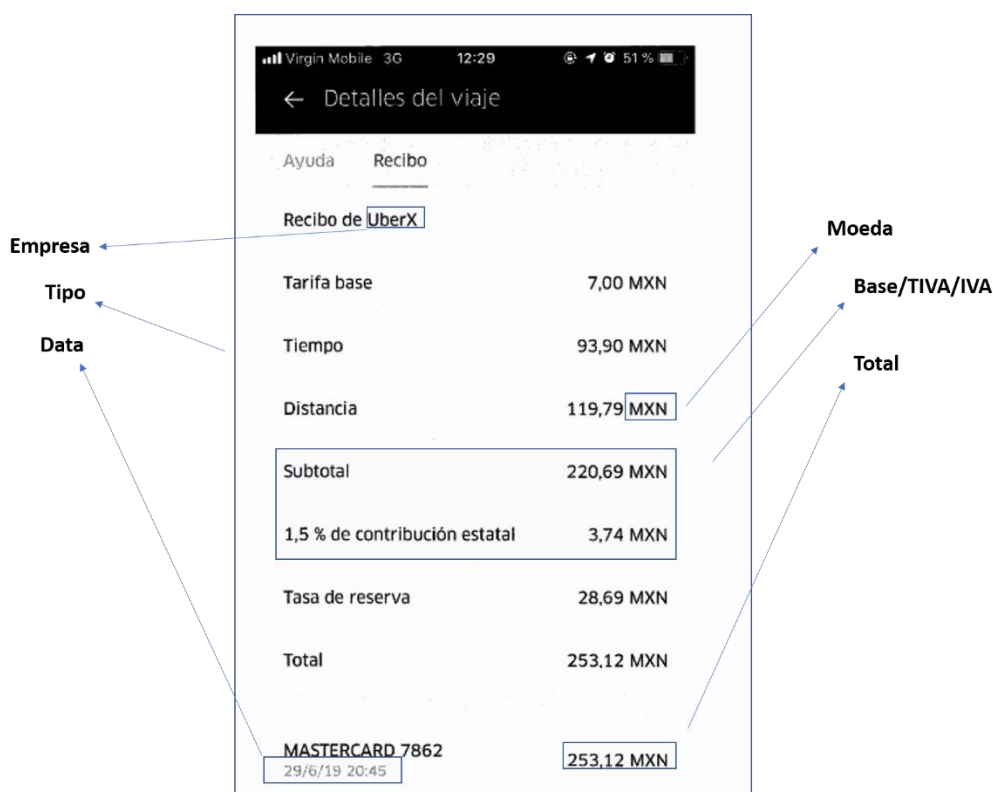


Figura 3.2: Exemplo da segunda fatura com quase todos os campos presentes

[illegible]

Como solução de continuidade foi equacionada uma segunda alternativa que não recorreria apenas aos serviços acima descritos na solução, mas seria assente em serviços de Deep Learning. Essa solução disponibiliza a melhoria contínua da solução na forma como deteta os campos, ou seja, quanto

mais faturas aparecerem, maior e melhor será a capacidade do modelo treinado em prever, com maior rigor, campos localizados em faturas diversificadas. Isto só é possível, porque com o passar do tempo será recolhido um maior volume de faturas que possibilitará a aprendizagem do modelo sobre as mesmas.

## **3.2 Seleção da Plataforma**

Conforme visto no Capítulo 2 foram analisadas diferentes plataformas e provedores de serviços que funcionam sobre a nuvem para identificar os serviços mais adequados para desenvolver a aplicação.

Em termos de orientação estratégica a MS decidiu que seria desejável concentrar o maior número de serviços num único provedor. Outro dos fatores que influenciou a seleção do provedor tem a ver com a MS ter uma linha de serviços de tecnologias de informação que ainda se encontra em desenvolvimento. Por isso, é desejável na área de IT testar serviços de provedores alternativos que possam ser implementados e, se robustos, virem a ser produtos vendáveis pela MS.

Assim, como já existia experiência interna com a Azure e com a Google Cloud, apesar das soluções destes provedores terem todos os requisitos necessários para o desenvolvimento da aplicação, a MS, após ponderar as diferentes alternativas, optou por seleccionar a Amazon Web Services.

Adicionalmente, para os serviços que a Amazon não dispunha ou não dava a resposta adequada para o desenvolvimento da aplicação, foram seleccionadas bibliotecas da linguagem de programação Python.

## **3.3 Especificações da Aplicação**

Este projeto dispõe de uma interface através da qual o utilizador introduz uma foto de uma fatura, iniciando os processos que executa em backend, referentes a operações de tratamento da mesma. A interface foi desenvolvida no programa Android Studio, com recurso à linguagem Kotlin. Os processos de tratamento de faturas recorreram à linguagem Python. Foi também decidida a utilização de uma base de dados NoSQL para o armazenamento da informação, pelas razões especificadas no ponto 2.2.7.

### 3.4 Diagrama de processos da solução com algoritmos de pesquisa de campos de faturas

Como foi referido no ponto 3.2, a MS definiu as linhas orientadoras e os requisitos para este projeto que incluíram a utilização de técnicas em DL, ML e NLP como base para o desenvolvimento da aplicação. Atendendo a que na fase inicial não existia um volume de faturas processadas elevado, foi decidido desenvolver a melhor solução possível dentro destas condições. Esta solução assenta em algoritmos que fazem procuras específicas sobre os dados, não gerando oportunidades de melhoria contínua.

Esta solução integra nove serviços da Amazon que são completados com o desenvolvimento de algoritmos e cujas principais atividades se sistematizam em oito passos, que partindo da imagem da fatura retornam os campos nela contidos. Esta solução baseou-se em quatro funções Lambdas porque tecnicamente a função Lambda apresenta limitação no tamanho do código e nas bibliotecas que podem ser instaladas, sendo necessário repartir a carga ao longo do processo. Os processos são apresentados na Figura 3.4:

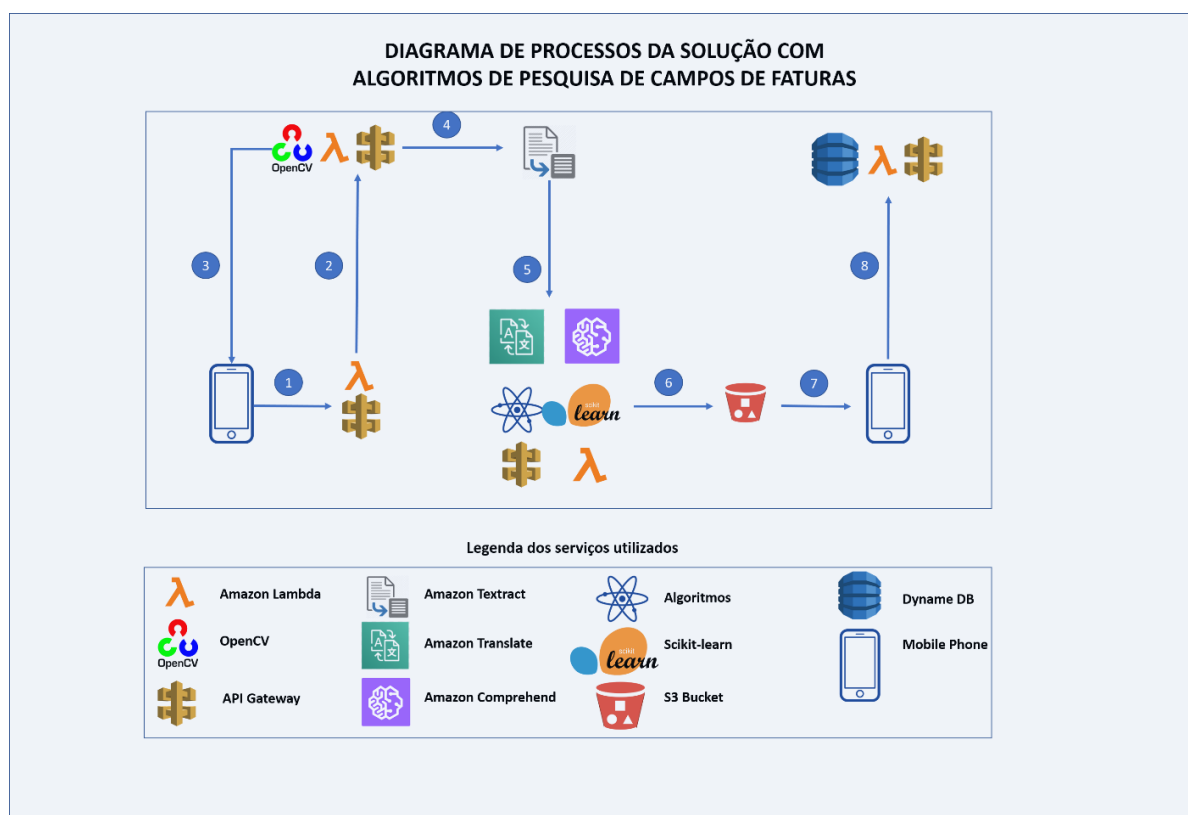


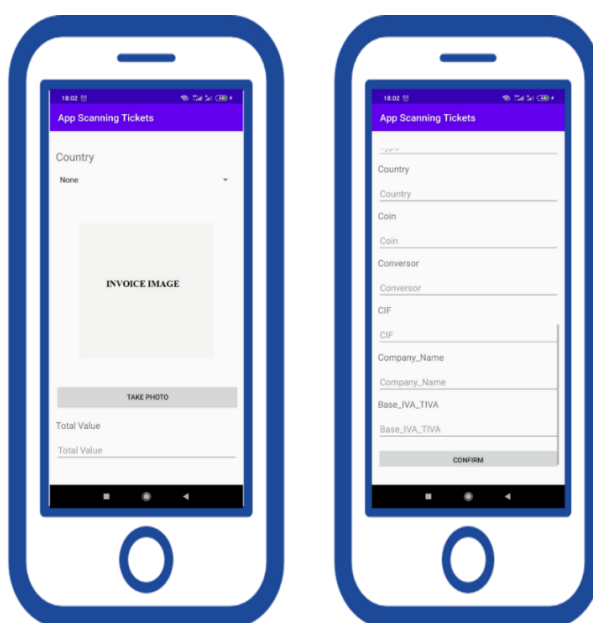
Figura 3.4 Diagrama de Processos da solução com algoritmos de pesquisa de campos de faturas

Apresenta-se sumariamente os oito passos desta solução que serão seguidamente detalhados, a partir do ponto 3.4.1, fazendo a descrição do trabalho realizado:

1. Envio da foto para processamento.
2. Validação da qualidade da fatura.
3. Retorno quando há má qualidade da imagem.
4. Extração do texto da fatura.
5. Tratamento dos dados e procura de campos.
6. Dados guardados na memória.
7. Retorno encontrado pelos algoritmos desenvolvidos.
8. Organização dos dados corretos finais numa base de dados.

### 3.4.1 Envio da foto para processamento

O utilizador tira uma foto com o seu telemóvel sendo possível identificar o país a que a fatura se refere, suportado numa lista de países possíveis, incluído a opção “None” para os casos dos países que não constam da lista e cuja identificação requer uma pesquisa feita por um algoritmo desenvolvido para esse efeito. Após a foto ser confirmada, os seguintes parâmetros são enviados: a foto em bytes, o país seleccionado pelo utilizador e um nome que é aleatório para referência futura da foto. O envio é feito para uma função Lambda, a partir da API Gateway, com o suporte de um método POST. A Figura 3.5 apresenta a vista da aplicação no momento em que o utilizador tira uma foto. Para o fazer é necessário pressionar no botão Take Photo e de seguida desencadeia-se o processo.



*Figura 3.5: Vista do utilizador antes do processamento de uma fatura*



### 3.4.2 Validação da qualidade da fatura

Outro requisito essencial é a aplicação conseguir reconhecer se uma foto tem qualidade ou se está desfocada. Para a automatização deste processo recorreu-se à biblioteca OpenCV na linguagem Python que permite desenvolver aplicações na área de visão computacional. A pesquisa efetuada permitiu concluir que uma técnica de identificação com precisão de imagens com baixa qualidade, é a variância de Laplace.

A técnica da variância de Laplace retira uma parte da imagem (usualmente em *grayscale*) e convolvê-la com o *kernel* 3x3 que pode ser observado na Figura 3.6. A razão pela qual este método funciona deve-se à definição do próprio operador Laplace, que é utilizado para medir a 2ª derivada de uma imagem. O operador Laplace destaca regiões de uma imagem que possuem mudanças rápidas de intensidade. Assim, pode ser utilizado para detetar *borders*. A validação da qualidade da foto está relacionada com o valor de variância que a mesma apresenta. Se a variância for elevada, ou seja, com uma grande variedade de respostas, significa que corresponde a uma imagem focada. Por outro lado, se a variância for muito baixa, significa que há uma pequena dispersão de respostas, indicando que a foto está desfocada.

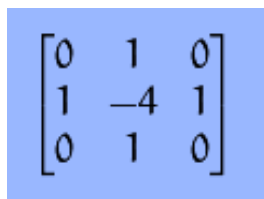

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Figura 3.6: O kernel de Laplace.

De seguida, é considerado um valor que estabelece um limite entre uma foto com qualidade e uma desfocada. A OpenCV permite realizar este processo com recurso a uma função, ou uma linha de código, sendo que a dificuldade reside em determinar o limite a partir do qual as imagens das faturas são consideradas válidas. Assim, através da utilização desta técnica, o utilizador é responsável por tirar uma foto com qualidade suficiente, para que os restantes processos se possam desenrolar e identificar os valores desejados. Este processo é executado a partir de uma chamada a uma função Lambda que desencadeia as etapas acima descritas. A Figura 3.7 evidencia uma fatura com qualidade suficiente e o valor da variância de Laplace correspondente à imagem:

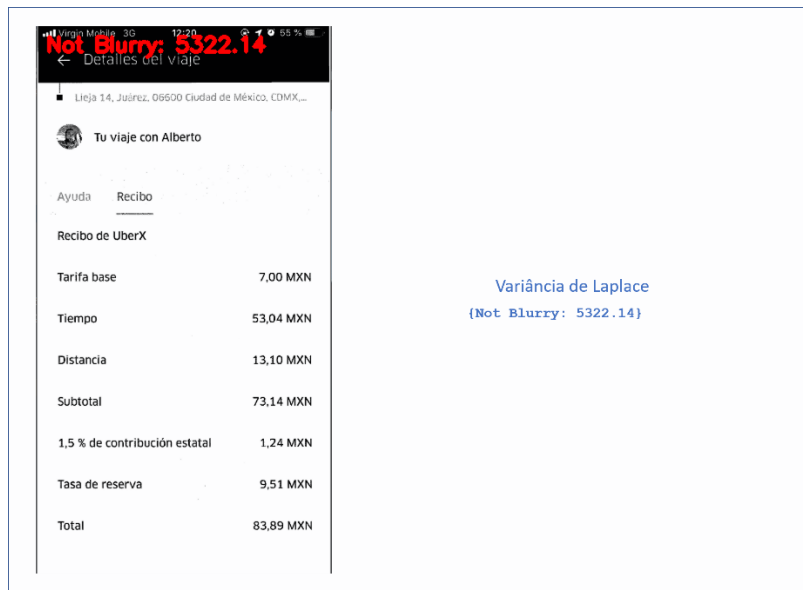


Figura 3.7 Foto retornada com o valor da variância de Laplace

### 3.4.3 Retorno quando há má qualidade da imagem

A decisão relativa à má qualidade da imagem teve por base o algoritmo mencionado no ponto anterior tendo sido definido o valor limite 100 para diferenciar uma foto com qualidade de uma foto desfocada. Se uma foto apresentar uma variância de Laplace com este valor ou menor, existe um retorno ao utilizador com a mensagem de que a foto não tem qualidade suficiente para ser processada, deste modo deverá repetir esta etapa, conforme se pode observar na Figura 3.8:

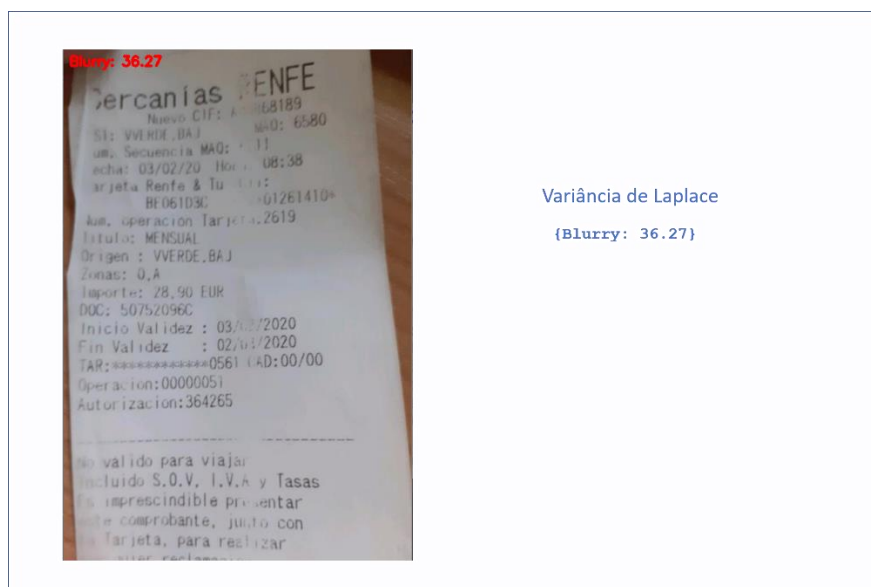


Figura 3.8 Foto considerada como desfocada

### 3.4.4 Extração do texto da fatura

Quando a fatura é validada, segue-se a extração do seu texto. Para que isto aconteça, é necessário recorrer ao serviço Amazon Textract que permite extrair as posições das palavras com coordenadas e texto correspondente. Deste modo, tem-se a informação recolhida pronta para ser processada e analisada nas etapas subsequentes.

### 3.4.5 Tratamento dos dados e procura de campos

Esta foi a etapa mais exigente na qual se desenvolveram mais processos e conexões entre os serviços. Para a simplificação geral do processo foi necessário proceder à normalização das faturas pelo tendo-se selecionado o idioma Inglês para conversão de todas as faturas que viessem a ser processadas. Para a execução deste processo utilizou-se o Amazon Translate que permite fazer a tradução das faturas. Com a questão linguística solucionada passou-se à fase da procura dos campos definidos pela MS.

Foram utilizadas expressões regulares com auxílio de métodos DL, ML e NLP providenciados por serviços da Amazon, combinados com algoritmos especificamente desenvolvidos para o efeito, conforme a Figura 3.9. O conteúdo de cada campo foi analisado e foram desenvolvidos algoritmos específicos de pesquisa, por forma a permitir fazer previsões mais precisas. Foram definidas as prioridades para escolha das técnicas a serem utilizadas colocando em primeiro lugar o DL, ML e NLP, seguidamente por expressões regulares e por fim quando não existissem estas alternativas procedia-se ao desenvolvimento de algoritmos específicos conforme fluxo da Figura 3.9:

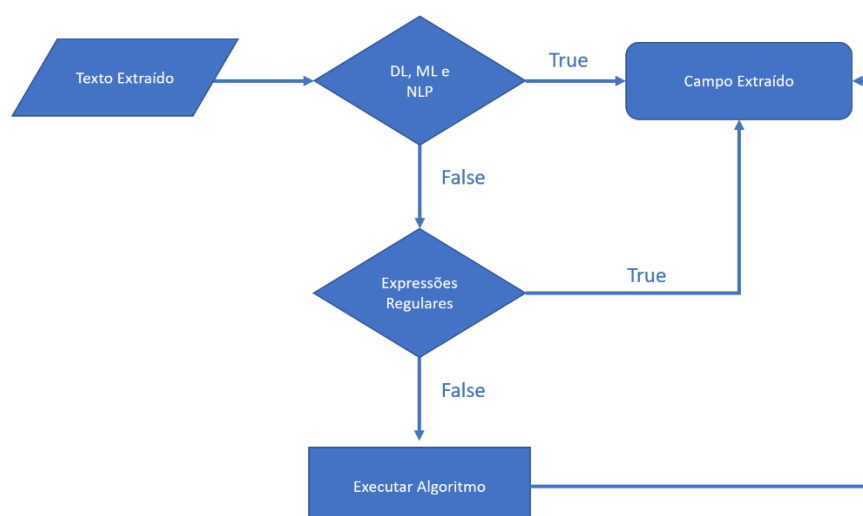
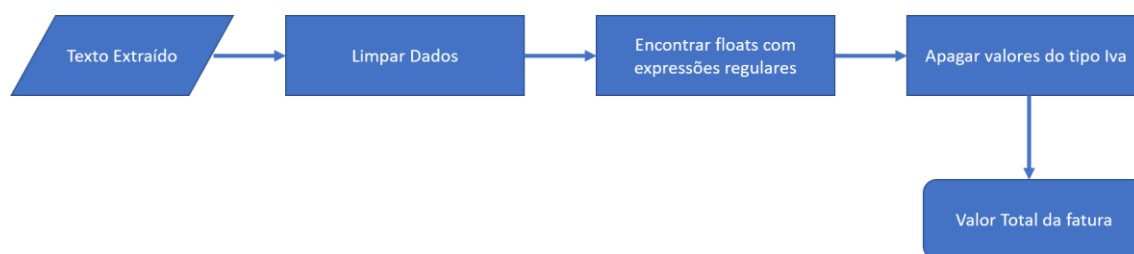


Figura 3.9 : Solução inicial equacionada para descobrir os campos necessários.

De seguida apresentam-se os algoritmos necessários para identificar os diferentes campos da fatura. O primeiro algoritmo a ser desenvolvido tinha o objetivo de conhecer o valor total de uma fatura. Apesar de parecer direto, a procura de valores numéricos em faturas, com recurso a expressões regulares, é uma tarefa complexa e difícil de se concretizar com sucesso. Isso deve-se à diversidade na disposição dos dados e à estrutura das faturas. Algumas condicionantes analisadas são descritas seguidamente, bem como as abordagens adotadas para superar as mesmas:

- Em Espanha e também em Portugal há faturas, que contêm o capital social da empresa, enquanto noutros países este dado não é apresentado. Neste caso, entendeu-se que o valor total deveria ser preferencialmente selecionado entre valores que se repetem, uma vez que o capital social aparece apenas uma vez.
- Se o cliente pagar com dinheiro, a fatura irá apresentar o valor que o cliente entregou para fazer o pagamento, que é normalmente, um valor superior ao valor total, podendo até incluir o valor entregue como troco. A solução a adotar é similar à anterior.
- O valor do IVA aparece sempre nas faturas e como vão ser considerados os valores decimais, se a fatura for de um valor inferior ao do IVA, por exemplo 5 euros, enquanto a taxa do IVA for de 13%, iria ser considerado o IVA como o valor total, uma vez que o valor absoluto 13 é superior ao valor 5. A solução adotada passou por excluir neste algoritmo os campos referentes ao IVA. Esta exclusão ocorre porque se recorreu a um algoritmo que utiliza a biblioteca Beautiful Soup, que pesquisa numa página web as tabelas atualizadas do IVA disponíveis em cada país.
- Os valores acima de 1000 euros apresentam muitas estruturas diferentes, relativas a vírgulas ou pontos para separação das unidades, por isso foi necessário modificar a forma como se faz a leitura desses números para que pudessem ser reconhecidos pelo algoritmo.

O algoritmo do cálculo do valor total é sumariado na Figura 3.10:

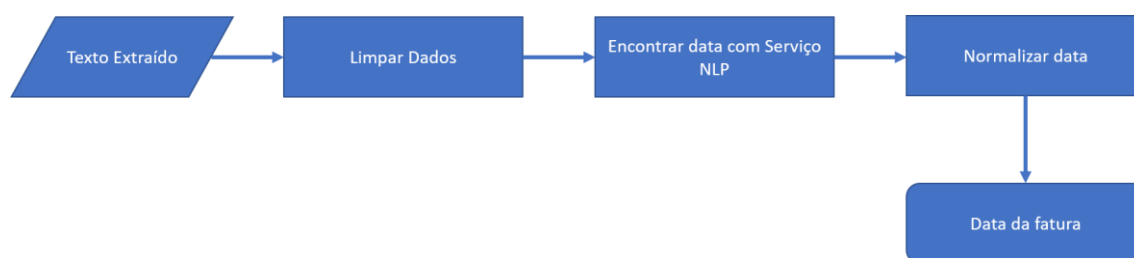


*Figura 3.10: Algoritmo desenvolvido para descobrir o valor total de uma fatura.*

Para criar o algoritmo que permita identificar a data recorreu-se ao serviço Amazon Comprehend que consegue detetar entidades.

Assim, foi feita uma recolha dos dados filtrando as entidades com o valor data, após esta filtragem foi feita uma busca sobre uma data modelo standard com o tipo DD/MM/YY. Se as datas encontradas não estivessem neste modelo fazia-se uma conversão automática para este formato. Caso contrário, retornava-se a data com maior número de caracteres.

O algoritmo do cálculo da data é apresentado na Figura 3.11:



*Figura 3.11: Algoritmo desenvolvido para descobrir a data de uma fatura.*

Outro algoritmo desenvolvido tem a ver com a tipologia das faturas, que a empresa pretendia ver estruturadas do seguinte modo:

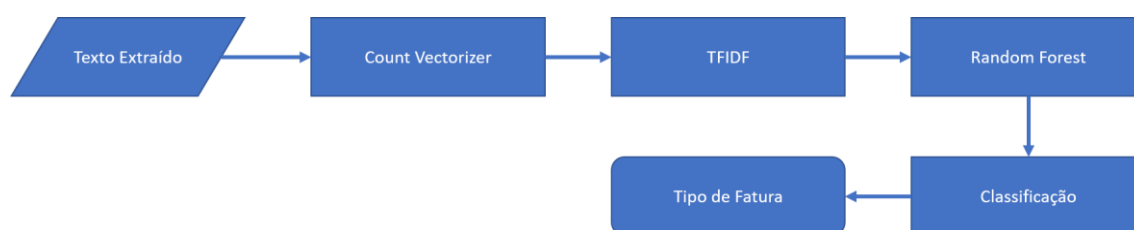
- Subsistence
- Tolls
- Taxi
- Mobile Phone
- Accomodation
- Undefined

Para esta categorização da tipologia das faturas começou-se por utilizar, o máximo de serviços da Amazon. Foi considerado o Amazon Comprehend que permite classificar em tempo real. Isto foi a opção inicial porque não são necessários conhecimentos de ML e tornava-se uma opção estável para novos programadores que viessem trabalhar no projeto. Todavia, os custos associados à manutenção deste modelo conectado são muito elevados, equivalendo a 44 euros/dia ou 16.000 euros/ano. Em função da análise de custos a MS solicitou que fosse desenvolvida uma solução interna mais económica e com bons resultados.

Na altura que se fez esta avaliação já tinham sido recolhidas pela empresa, cerca de 4000 faturas. O dataset compreendeu dois parâmetros, o texto da fatura traduzido para a língua inglesa e os

diferentes tipos de faturas definidas. O carregamento destas faturas foi feito manualmente e dado que apresenta o tipo de fatura correto no dataset, foi possível adotar modelos de ML supervisionados.

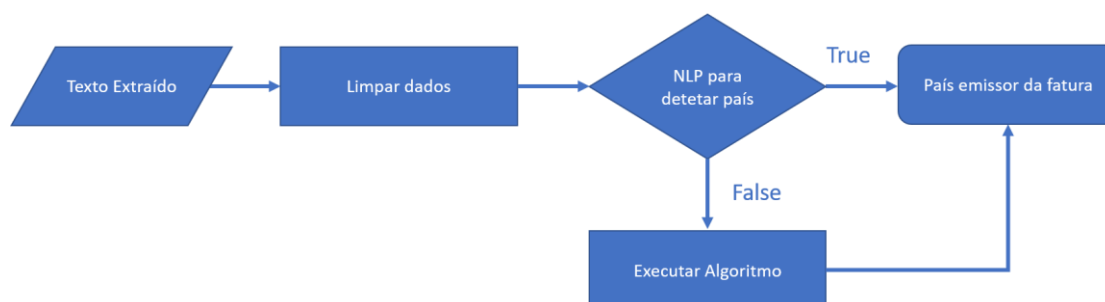
A biblioteca em Python adotada para o desenvolvimento desta solução foi a Sklearn e o processo consistiu nas seguintes etapas: utilizou-se o dataset anteriormente descrito, converteram-se os textos numa matriz de tokens recorrendo ao Count Vectorizer, fez-se uma conversão a partir do TFIDF que indica a importância de uma palavra num documento em relação a um conjunto de documentos, seguidamente considerou-se o modelo de classificação Random Forest por ter correspondido ao esperado e por fim foi realizada a classificação que identifica o tipo de fatura. O modelo sumariza-se na Figura 3.12:



*Figura 3.12: Algoritmo desenvolvido para descobrir a data de uma fatura.*

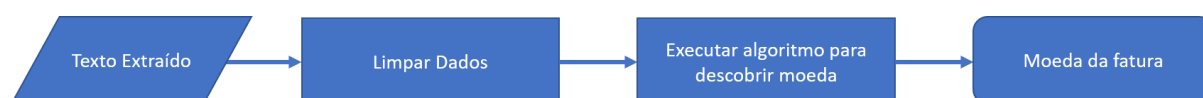
No ponto 3.4.1 é referida a possibilidade de o utilizador introduzir um input para seleccionar o país de origem da fatura. Também, o utilizador pode seleccionar da lista de países o input “None”, que se refere a um algoritmo específico desenvolvido, que é processado para descobrir um país que não conste na lista. Este algoritmo utiliza o serviço Amazon Comprehend, contendo a funcionalidade de poder identificar localizações num conjunto de texto, se não for encontrado nenhum país ou cidade, pesquisando localizações específicas que detete na fatura. Considerou-se esta opção de input porque existe a possibilidade de o país de origem da fatura não constar na mesma.

Quando o utilizador selecciona o país gerando o input realiza uma atividade essencial, dado que existem campos que são dependentes da localização, como os IVA's e a moeda. Assim, considera-se o país como o campo mais determinante para pesquisa deste ponto. O processo do algoritmo sumariza-se na Figura 3.13:



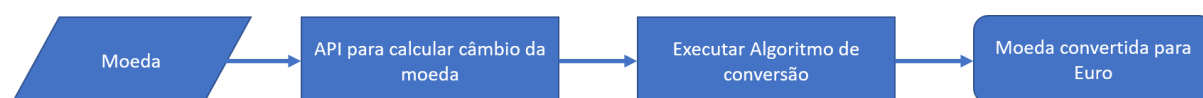
*Figura 3.13: Algoritmo desenvolvido para descobrir o país da fatura.*

De seguida, foi desenvolvido um algoritmo que faz a procura da moeda da fatura. Se o país correto for seleccionado no Input existe a referência direta ao tipo de moeda. Não sendo esse o caso, foi desenvolvido um algoritmo que faz buscas, com base, em símbolos e nomes de moedas, conforme descrito na Figura 3.14:



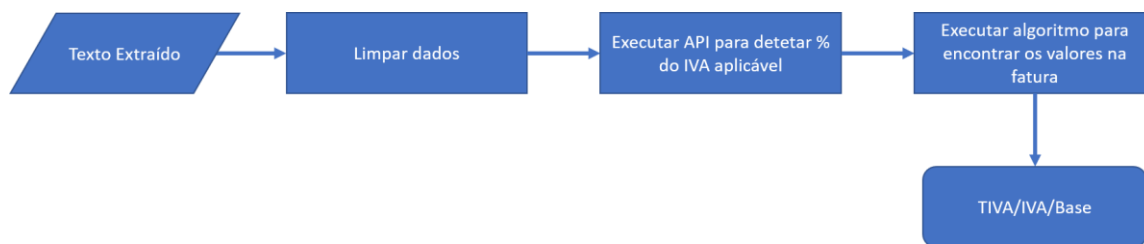
*Figura 3.14: Algoritmo desenvolvido para identificar a moeda da fatura.*

Após a deteção da moeda é necessário fazer uma normalização, pelo que se decidiu que o Euro seria a moeda principal da aplicação. Isto significa que se uma moeda for, por exemplo, o US Dólar, existirá uma conversão para Euro e serão retornados ambos os valores. Esta conversão é realizada com recurso a uma API, que retorna a conversão de todo o tipo de moedas. Isto permite a conversão do valor total da fatura em euros, conforme descrito na Figura 3.15:



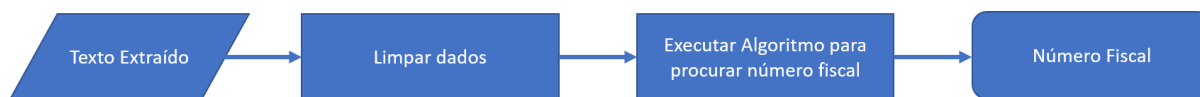
*Figura 3.15: Algoritmo desenvolvido para converter o a moeda da fatura para euros.*

Outra procura necessária em faturas é a identificação dos valores relacionados com a entidade fiscal do país, nomeadamente o IVA. Para o cálculo tem que se identificar a base tributável, a taxa aplicável ao valor do IVA de cada produto, os totais líquidos parciais de acordo com as taxas aplicáveis e IVAs parciais. Tal como aplicado para o cálculo do valor total, utilizou-se a API que permite ir buscar os valores do IVA de vários países a uma página web. Desta forma, é possível comparar os valores das faturas com os valores retornados pela API. Após esta comparação são feitos cálculos matemáticos através do algoritmo que faz validações, entre os valores da fatura e os valores reais pelas fórmulas do algoritmo, dando o retorno dos valores corretos. A Figura 3.16 apresenta as etapas deste processo:



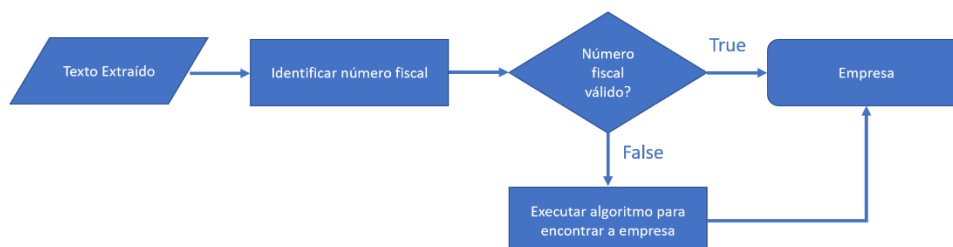
*Figura 3.16: Algoritmo desenvolvido para descobrir o Tipo Impositivo (% específica por produto), Base Tributável e IVA da fatura (global e parcial).*

Outros dos valores a retornar tem a ver com a correspondência do número fiscal. Para a sua deteção foi desenvolvido um algoritmo com recurso a expressões regulares, pois em cada país o número de contribuinte é sempre apresentado no mesmo formato. Para cada país será necessário desenvolver, tal como foi feito para Espanha, um algoritmo que identifique o formato tipo do número de identificação fiscal do país. Apresenta-se o algoritmo na Figura 3.17:



*Figura 3.17: Algoritmo desenvolvido para descobrir o número fiscal de uma fatura.*

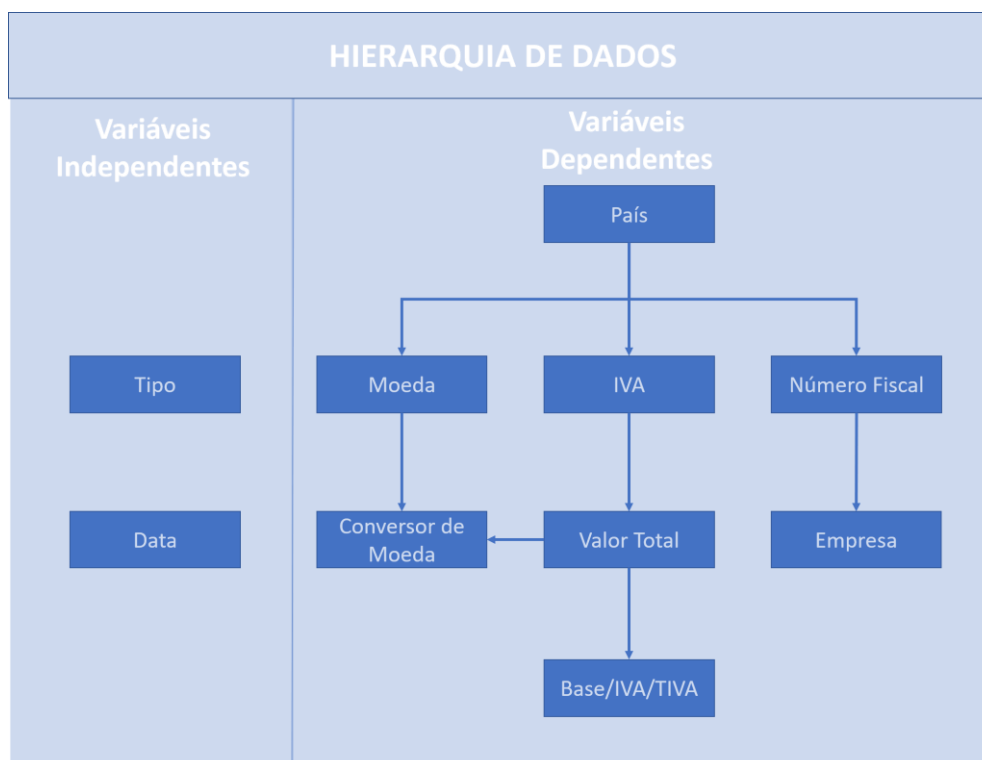
Por fim, foi desenvolvido um algoritmo que permite identificar a empresa que emitiu a fatura. Parte-se do número de identificação fiscal e com uma API envia-se um pedido com esse número a um website que possui a base de dados de todos registos fiscais das empresas. Neste momento esta solução é apenas aplicável a empresas com registo em Espanha. Se não existir esta correspondência, foi feito um algoritmo que procura o nome da empresa que consta na fatura, sendo este o valor retornado, conforme se pode ver na Figura 3.18.



*Figura 3.18: Algoritmo desenvolvido para descobrir a Empresa.*



Após a identificação dos dados a serem processados e a forma como se iria tratar cada campo, foi feita a hierarquização e dependências dos vários campos. Como tinha sido descrito anteriormente, o dado país é considerado como o mais importante numa fatura. Partindo da identificação do país, é possível encontrar a moeda, o IVA e o número fiscal que constam na fatura. Cada um destes campos, por sua vez, tem outros dados dependentes como o conversor de moeda, a empresa e o valor total. As dependências poderão ser unitárias ou combinadas entre os dados. Por outro lado, os campos data e tipo são completamente independentes. A Figura 3.19 apresenta a hierarquia dos dados.



*Figura 3.19 Hierarquia dos campos a encontrar numa fatura*

### 3.4.6 Dados Armazenados

Cada vez mais, a informação é importante para as organizações e a sua relevância pode mudar ao longo do tempo. Nesse sentido, há todo o interesse em selecionar os dados críticos associados ao processo de faturas e proceder ao seu armazenamento. Neste projeto foram selecionados como sendo críticos manter os seguintes dados:

- Foto original da fatura - ficheiro tipo jpg.
- Campos encontrados pelo processo – ficheiro tipo json.
- Texto original da fatura – ficheiro tipo txt.
- Texto traduzido da fatura – ficheiro tipo txt.

- Palavras com respetivas coordenadas e índice de confiança apresentado pelos algoritmos da Amazon – ficheiro tipo csv.

Esta informação é armazenada sempre que um utilizador tire uma foto e o processo é executado do início ao fim. Foi decidido que a melhor solução para armazenamento destes dados seria um S3 Bucket porque permite guardar um elevado número de dados, assim esta opção pode-se considerar uma solução de Big Data. A informação pode ser considerada como um repositório de dados que podem vir a ser utilizados em processos mais sofisticados de Deep Learning e Machine Learning no futuro para este e outros fins.

### 3.4.7 Retorno Devolvido pelos Algoritmos Desenvolvidos

Após a conclusão do processo, é enviada para o utilizador uma resposta que contém os dados que foram possíveis detetar durante a realização das atividades acima descritas. Existe também a possibilidade de o utilizador corrigir dados incorretos que foram recebidos substituindo-os por dados reais. Na Figura 3.20 é apresentado um exemplo da vista da aplicação no momento do processamento dos dados.

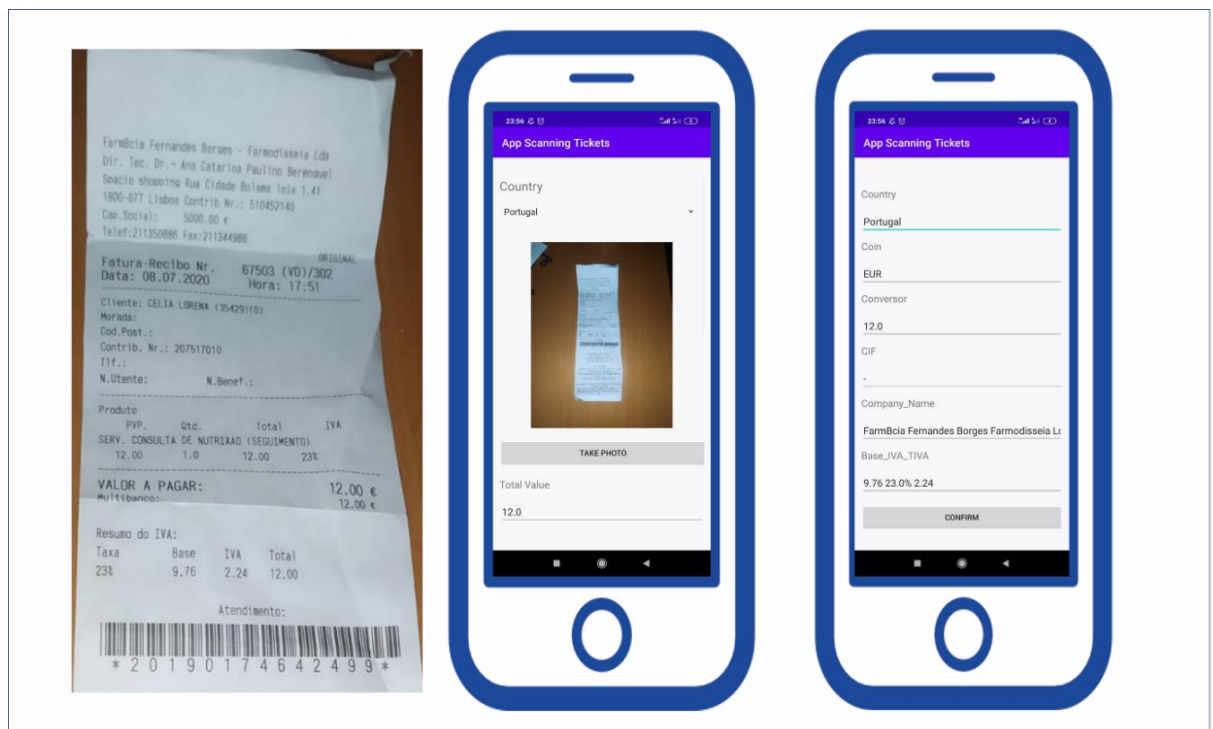


Figura 3.20: Output de uma fatura com os dados encontrados e fatura correspondente.

### 3.4.8 Organização dos Dados Corretos Finais numa Base de Dados

Quando o utilizador dispõe dos dados corretos procede ao seu armazenamento, a partir da seleção do botão confirm, da figura anterior. Este armazenamento é feito para uma tabela de base de dados NoSQL, concretamente a Dynamo DB. Na Tabela 3.2 é apresentado um exemplo de dados que são guardados na Dynamo DB e que pela extensão da tabela não são totalmente reproduzidos:

| Filename                      | Valor Total | Data                   | ... |
|-------------------------------|-------------|------------------------|-----|
| Fatura_01_02_07_2020_19:00:20 | 9.12        | 25 de Novembro de 2019 | ... |
| Fatura_02_02_07_2020_19:00:22 | 12.00       | 20/5/2020              | ... |
| Fatura_03_02_07_2020_19:00:22 | 16.25       | 3/6/2019               | ... |

*Tabela 3.2 Representação dos dados que ficam agrupados na Base de Dados Dynamo*

## 3.5 Diagrama de Processos da Solução com Modelo de Deep Learning

A segunda solução tem os processos *core* iguais aos da primeira solução. Estes processos incorporam a melhoria contínua, dado que incluem um modelo de Deep Learning que, à medida que vai acumulando mais dados, vai evoluindo e aprimorando a sua capacidade de resposta no tratamento de faturas.

A segunda solução é integrada com a utilização de uma instância EC2 substituindo duas funções Lambda e faz a deteção dos campos com recurso a uma rede convolucional sendo estas as grandes diferenças relativamente à solução inicial. A EC2 está desenvolvida com a biblioteca Flask que serve de API para a aplicação. Adicionalmente, os serviços Gunicorn e Nginx oferecem suporte à aplicação Flask desenvolvida.

Neste cenário foi equacionada a incorporação de uma solução que interpretasse a imagem da fatura e retornasse as coordenadas correspondentes a cada campo a tratar. Uma técnica que é vulgarmente utilizada para solucionar situações equiparadas, é o reconhecimento de objetos numa imagem, funcionando sobre redes convolucionais. A título de exemplo, usa-se para diferenciar um cão de um gato numa imagem. Assim, considerou-se esta técnica com as devidas adaptações para solucionar a identificação das coordenadas dos campos de texto.

Para o desenvolvimento desta solução, existiu a necessidade de construir um dataset que tivesse como input as imagens e output as posições das coordenadas correspondentes. O conjunto de faturas apresentado pela MS corresponde a uma amostra de 10.000 unidades, no entanto, após uma limpeza

dos dados, o dataset foi ajustado para 2000 faturas. Existiu este ajustamento porque o dataset foi limpo pela eliminação das faturas manuscritas, das faturas repetidas e das faturas que não tinham qualidade suficiente. Assim, criou-se uma solução que permite que sempre que seja necessário acrescentar faturas ao dataset, o processo executa a operação de forma automatizada.

Apesar de se ter desenvolvido a segunda solução, a mesma apenas virá a ser utilizada em pleno quando existir capacidade interna na empresa de processamento de um número significativo de faturas. Desta forma, será possível testar a eficácia desta solução comprovada com dados, dando garantias à empresa que lhe permita pôr em produtivo esta solução. Uma vez que esta segunda solução ainda é um teste adotou-se a procura dos campos valor total e data porque são estes campos que existem em todas as faturas e aos quais faz mais sentido aplicar as redes convolucionais. Deste modo, decidiu-se desenvolver a mesma interface, mas com a diferença de que apenas apareceriam esses dois campos, em vez dos oito da primeira solução.

O diagrama de processos da segunda solução consta na Figura 3.21. Dado haver semelhanças em ambas as soluções o detalhe das várias etapas será apenas feito nos casos em que existem modificações no processo de execução:

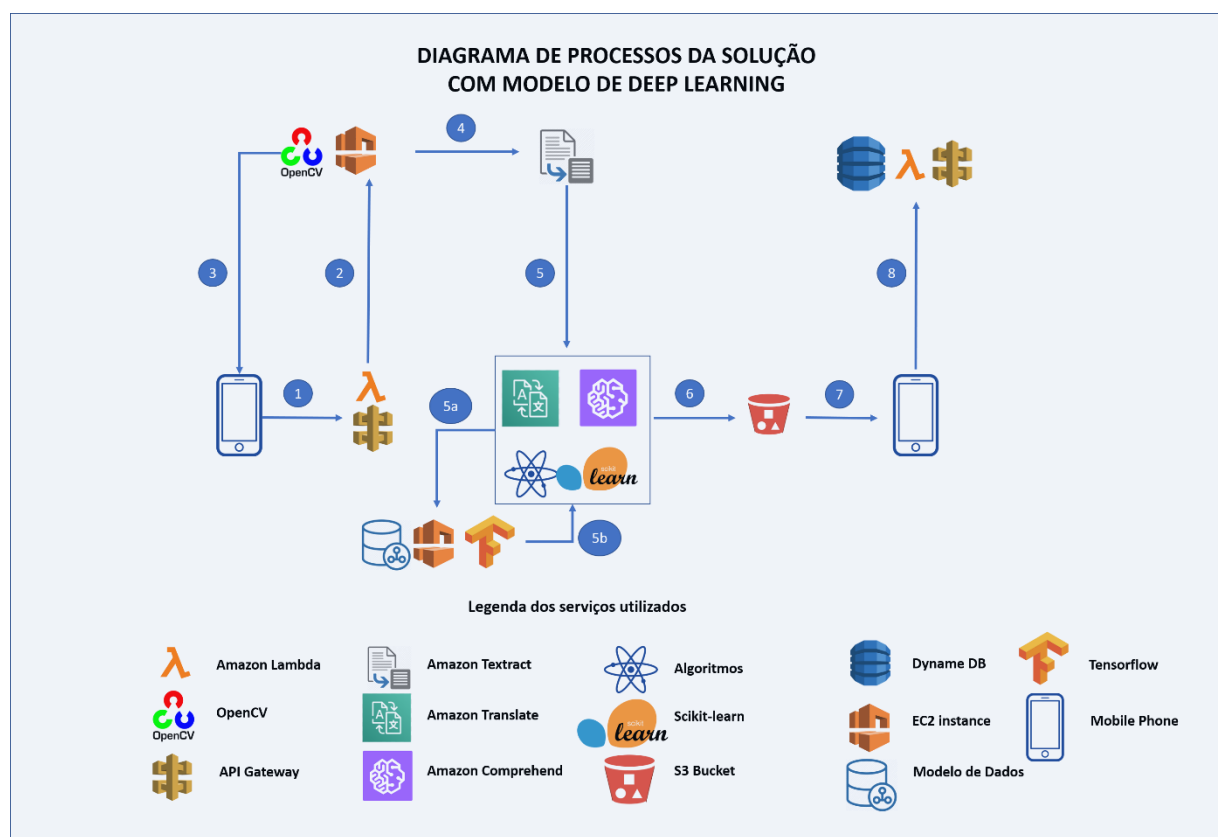


Figura 3.21 Diagrama de Processos da solução com Deep Learning

Apresenta-se sumariamente os oito passos desta solução dos quais apenas serão descritos o 5a e 5b, a partir do ponto 3.5.1, pois os restantes tal como referido acima são iguais aos da primeira solução com ligeiros ajustamentos já referidos no enquadramento da mesma:

1. Envio da foto para processamento.
2. Validação da qualidade da fatura.
3. Retorno quando há má qualidade da imagem.
4. Extração do texto da fatura.
5. Limpeza e Tratamento dos dados.

**5a Identificação das coordenadas da imagem – Dataset e Treino do Modelo.**

**5b Retorno das coordenadas e processamento dos algoritmos.**

6. Dados guardados na memória.
7. Retorno encontrado pelos algoritmos desenvolvidos.
8. Organização dos dados corretos finais numa base de dados.

### **3.5.1 Identificação das coordenadas da imagem – Dataset e Treino do Modelo**

A identificação das coordenadas da imagem foi dividida em duas etapas: o desenvolvimento de um dataset que permitisse a estruturação dos dados a treinar e seguidamente fez-se um treino do modelo para retornar as coordenadas do Valor Total e Data, que se passa a detalhar seguidamente.

#### **Criação do Dataset**

A primeira parte deste processo consistiu no desenvolvimento de um dataset que pudesse ser utilizado para treino e posterior previsão dos campos em faturas. Dado que se decidiu utilizar uma rede convolucional que faz deteção de objetos em imagens, é necessário fornecer coordenadas ao dataset para que a rede convolucional possa identificar a localização exata das caixas a encontrar. Para este efeito, foi decidido que o dataset iria conter oito parâmetros:

- Filename – Nome da imagem.
- Width – Largura da imagem.
- Height – Altura da imagem
- Class – Tipo correspondente a encontrar dentro de uma fatura, por exemplo, valor total.
- Xmin – Posição inicial da coordenada x no eixo.
- Ymin – Posição inicial da coordenada y no eixo.

- Xmax – Posição final da coordenada x no eixo.
- Ymax – Posição final da coordenada y no eixo.

Na Figura 3.22 tem-se uma vista das coordenadas que foram consideradas como as respostas corretas para o modelo de DL. São definidos pelos pontos Xmin, Xmax, Ymin e Ymax, sendo agrupadas com o texto encontrado, a altura e largura da imagem e por fim a classe correspondente.



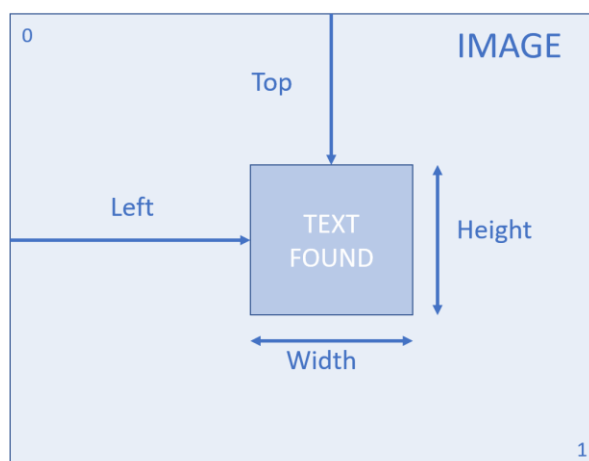
*Figura 3.22: Representação das coordenadas consideradas existentes numa fatura*

Todas as imagens incluídas no dataset seguiram o mesmo processo, ou seja, as caixas que eram criadas correspondiam a valores numéricos, concretamente às posições do x e y. Para isso, foi necessário desenvolver um método para processar cada imagem. Um dos processos possíveis analisados, consistiu no recurso à biblioteca LabelImg, que é desenvolvida em Python e permite aos utilizadores etiquetarem imagens de uma forma simples e direta. Estas imagens ficam guardadas no formato desejado e no formato xml. Apesar desta ser uma boa solução, tem um problema, que é a necessidade de processar as imagens manualmente, o que requer que uma pessoa despenda tempo a processar cada imagem. Dado que a MS não estava disposta a despendar recursos para este fim, foi desenvolvida uma solução automatizada, no âmbito do projeto.

Esta automatização iniciou-se com a seleção das fotos válidas, de entre a amostra disponível de 10.000 imagens. A tipologia de erros encontrada, que levou à exclusão das imagens, tem a ver com a qualidade da foto, a algumas serem escritas à mão e difíceis de processar e outras estarem repetidas.

Após a obtenção de uma foto, o utilizador recorta as margens da imagem, ficando apenas com os dados necessários para serem processados. Tal como referido anteriormente, a informação relevante é extraída, nomeadamente as coordenadas juntamente com o texto.

O Amazon Textract possibilita definir a posição de cada palavra, fazendo a extração das mesmas, que ficam agrupadas em quatro coordenadas diferentes. O top é a posição que vai desde o limite superior de uma imagem até ao limite superior da caixa com o texto encontrado, o left é a posição que vai do limite lateral esquerdo até ao lado esquerdo da caixa, o width é o tamanho da largura da caixa e o height é o tamanho da altura da caixa. Estas coordenadas encontram-se normalizadas e estão contidas em valores entre 0 a 1. Estas referências constam na Figura 3.23. Para cada coordenada e texto correspondente é identificado o grau de confiança que permite detetar a precisão de certeza relativamente à informação encontrada.



*Figura 3.23: Representação das coordenadas que o Amazon Textract identifica para cada texto encontrado*

Após a execução desta etapa, o utilizador valida os dados fazendo ajustamentos, se necessário, e a informação com os dados corretos é armazenada. Foi desenvolvido um script que extrai a informação da base de dados e os dados que tinham sido guardados num bucket. A associação dos ficheiros é feita com base no nome da imagem, que existe quer no Dynamo DB, quer no S3 Bucket. Apesar de ser uma solução desenvolvida à medida, a técnica associada à Amazon Textract pode não ser sempre eficaz e haver alguns erros gerados por uma leitura incorreta dos dados, pelo que se recorreu à distância Levenshtein para superar este problema.

A distância Levenshtein permite calcular o número mínimo de operações necessárias para transformar uma string noutra string. Quanto maior o valor que atribuirmos a essa distância, maiores são as possibilidades de existirem correspondências entre as palavras a identificar. Considerando um

conjunto de palavras, esta distância permite identificar a palavra a descobrir comparando com a palavra mais semelhante existente no dataset, permitindo fazer correspondências e diminuir erros.

O dataset desenvolvido resulta do tratamento de múltiplas faturas, seguindo o processo acima descrito e conforme a Figura 3.24 abaixo.

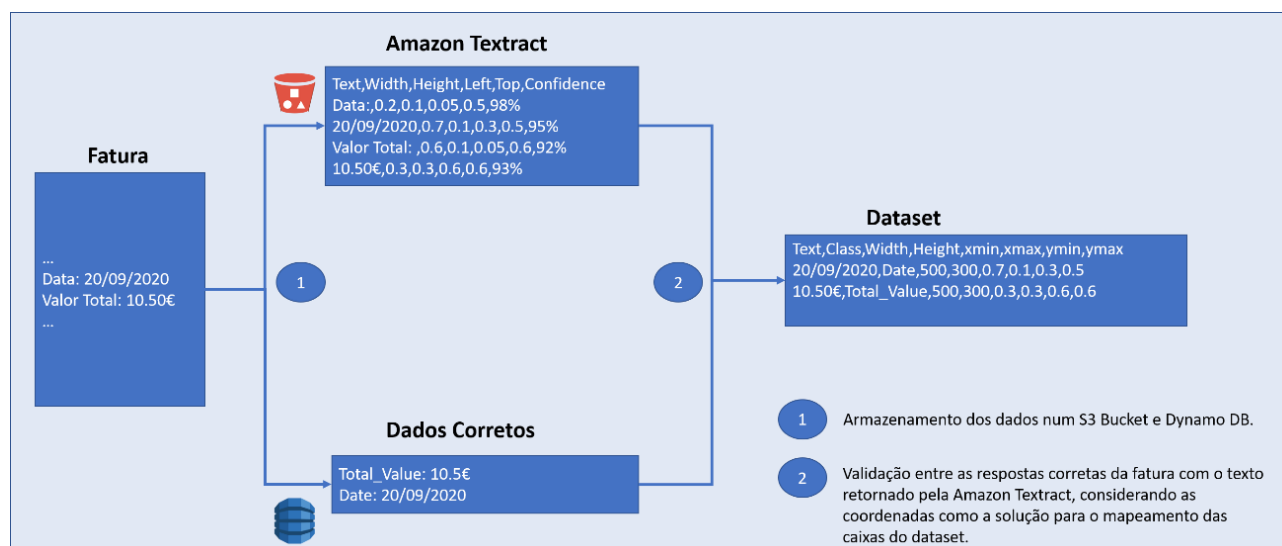


Figura 3.24: Processo para o desenvolvimento do Dataset gerado automaticamente

## Treino do Modelo em Deep Learning

Para fazer a deteção de objetos em imagens foram analisadas várias soluções, destacando a Amazon SageMaker, a Amazon Rekognition e uma solução com bibliotecas de Deep Learning, referido na Tabela 2.2. Apesar dos resultados dos dois primeiros serviços corresponderem a uma boa resposta às necessidades da aplicação, testados através de provas de conceito, os custos destes serviços são muito elevados. Por esse facto, foi desenvolvida uma solução inspirada no modelo de funcionamento da Amazon Rekognition mas com uma biblioteca de Deep Learning que foi treinada numa máquina local, sendo deste modo uma solução economicamente mais vantajosa.

A biblioteca de Deep Learning incorporada, denominada Tensorflow, é suportada pela API Tensorflow Object Detection. Uma das dificuldades que é comum nos modelos de deep learning é de serem capazes de localizar e identificar múltiplos objetos de uma forma precisa. A solução assente na API escolhida facilita a construção, o treino e a deteção de objetos. As principais etapas do desenvolvimento desta solução incluíram:

1. Instalação do ambiente de desenvolvimento Anaconda para ser possível utilizar seguidamente um ambiente virtual com Python 3.6 em conjunto com bibliotecas específicas. Inclui-se nesta



etapa a instalação de ferramentas para se poder treinar o modelo numa GPU, nomeadamente o CUDA e cudNN.

2. Cópia da API Tensorflow Object Detection para o repositório da aplicação com algumas alterações, como a criação de uma pasta images, na qual ficaram guardados o dataset com as imagens correspondentes.
3. Utilização de um modelo pré-treinado, para fazer face à insuficiência de dados, permitindo obter melhores resultados no treino. A Tensorflow apresenta uma série de possibilidades relativamente a modelos pré-treinados, em que milhares de imagens foram utilizadas para treino, variando os modelos em velocidade versus precisão. Este modelo pré-treinado apresenta um conjunto de imagens como carros, pessoas, animais, objetos, entre outros. Procurou-se um modelo que contivesse números ou letras no seu conjunto de dados, mas não foi encontrado esse modelo ideal, assim procedeu-se à utilização do Faster-RCNN-Inception-V2 e a deteção funcionou eficazmente, se bem que a velocidade fosse mais baixa.
4. Geração da informação para treino e teste.
5. Inicialização do processo de treino utilizando o pipeline do modelo pré-treinado, modificando os dados a serem analisados.
6. Treino do modelo durante um período de tempo, que permitisse a aprendizagem do modelo no reconhecimento dos campos nas faturas.
7. Análise dos resultados, identificando o momento em que o modelo entra em overfitting e não existe mais necessidades de continuar o seu treino, atingindo o melhor resultado.
8. Exportação do gráfico de inferência e fazer as previsões dos dados a encontrar nas faturas com base no modelo treinado.
9. Retorno das coordenadas e classes identificadas para cada uma das caixas encontradas.

Este modelo foi guardado numa instância EC2, o que permite que esteja sempre conectado, existindo a possibilidade de aceder ao mesmo sempre que surjam novas faturas. A melhoria contínua está dependente do treino do modelo que incorpore dados adicionais que tenham sido processados.

Para demonstrar o processo acima descrito, apresenta-se a Figura 3.25 de uma fatura e o retorno das coordenadas correspondentes às caixas de cada um dos campos encontrados.

As coordenadas retornadas são constituídas pelos valores falados na Figura 3.22, concretamente o Xmin, Ymin, Xmax e Ymax. A Figura 3.25 apresenta os retornos:

- Valor Total, tendo as coordenadas [0.65,0.76,0.66,0.86] e um índice de confiança de 99%.
- Valor Total, tendo as coordenadas [0.30,0.77,0.32,0.87] e um índice de confiança de 98%.
- Data, tendo as coordenadas [0.10,0.73,0.12,0.88] e um índice de confiança de 99%.

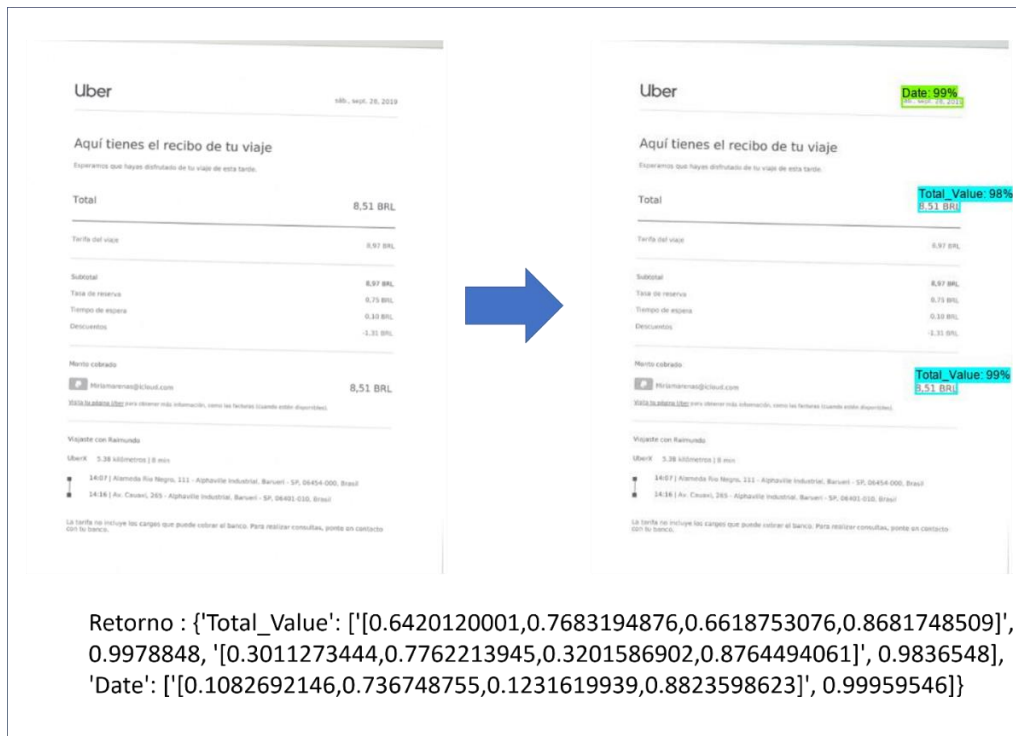


Figura 3.25: Coordenadas retornadas pelo modelo de Deep Learning desenvolvido por uma fatura da Uber

### 3.5.2 Retorno das coordenadas e processamento dos algoritmos

Seguidamente, é necessário fazer uma associação entre as coordenadas recebidas pela aplicação e as coordenadas que o Amazon Textract identificou anteriormente, com o objetivo de normalizar as coordenadas, encontrando o ponto central de caixa. Após a detecção destes pontos centrais são calculadas distâncias euclidianas entre os pontos centrais e é considerado como caixa correspondente a que contiver o valor inferior, conforme a Figura 3.26.

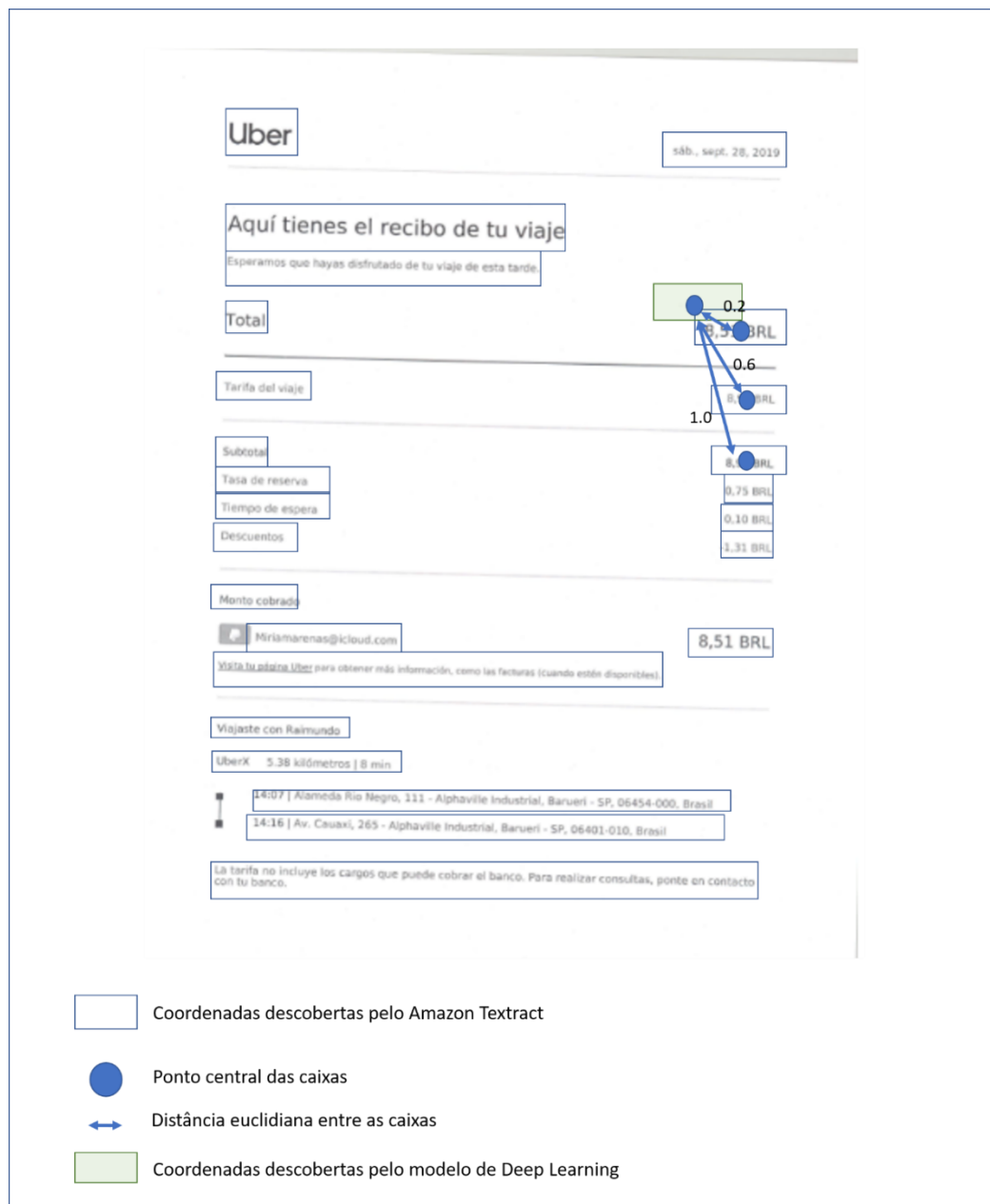
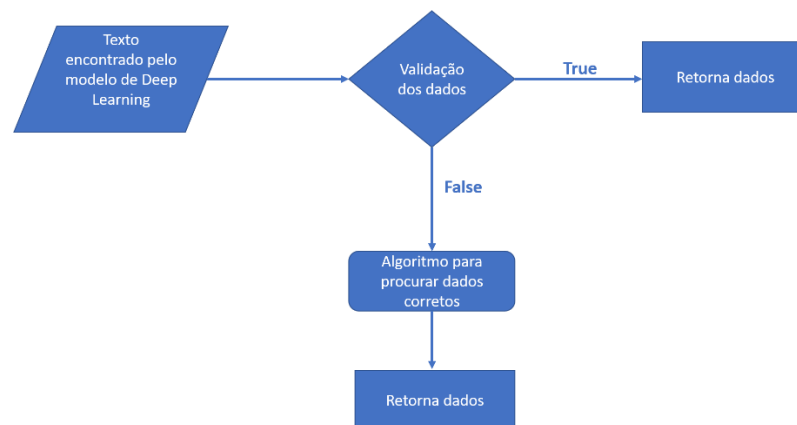


Figura 3.26: Correspondência entre as coordenadas enviadas pelo modelo e as coordenadas identificadas pelo Amazon Textract retornando o texto que apresenta a menor distância

Também existe a necessidade de fazer a validação da fiabilidade da informação retornada pelo algoritmo de Deep Learning, a fim de reconhecer o valor correto. Se o mesmo não tiver sido retornado, como por exemplo a data, deve-se recorrer a outro processo que identifique a mesma. Estas validações são efetuadas com os algoritmos definidos na primeira solução, conforme descrito no ponto 3.4.5. A título de exemplo, é necessário fazer esta validação pois podem surgir casos em que o valor retornado pela *bounding box* com valor data ser o que corresponde ao valor total da fatura. Com estas validações existe a possibilidade de se fazerem correções e o retorno de um valor correto.

A validação dos dados comprova se o tipo de dados encontrado está correto como, por exemplo, se o valor total retornado é um número. De seguida, se for verdadeiro considera-se esse retorno correto, caso contrário é executada a primeira solução que faz uma procura mais específica do campo sobre o texto retornado pelo Amazon Textract. Desta forma, existe uma junção entre as duas soluções que faz com que sempre que os valores retornados não forem considerados como o campo a procurar, a primeira solução é executada e é retornado o valor encontrado.



*Figura 3.27 Representação do algoritmo que faz a validação dos dados que são retornados pelo modelo de DL*

## Capítulo 4

### Resultados e Avaliações

Neste capítulo vão-se apresentar os resultados obtidos com os modelos e algoritmos desenvolvidos. Sistematizam-se as análises aos resultados do modelo desenvolvido que classifica o tipo de fatura, aos resultados obtidos pelos algoritmos de procura e também a avaliação do modelo desenvolvido em Deep Learning.

#### 4.1 Resultados do modelo para identificar tipologia de fatura

Existiu a necessidade de desenvolver um modelo próprio em sklearn que fizesse a identificação do tipo de fatura, baseando-se no texto que a mesma continha. Como se pode ver na Tabela 4.1, a precisão média ponderada foi de 0.98, havendo quatro tipos de faturas em que se obteve uma precisão máxima – Accomodation, Mobile Phone, Taxi e Tolls.

|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| Accomodation | 1.00      | 1.00   | 1.00     | 61      |
| Mobile Phone | 1.00      | 1.00   | 1.00     | 4       |
| Subsistence  | 0.97      | 1.00   | 0.99     | 523     |
| Taxi         | 1.00      | 0.99   | 0.99     | 214     |
| Tolls        | 1.00      | 0.67   | 0.80     | 3       |
| Undefined    | 0.98      | 0.80   | 0.88     | 59      |
| accuracy     |           |        | 0.98     | 864     |
| macro avg    | 0.99      | 0.91   | 0.94     | 864     |
| weighted avg | 0.98      | 0.98   | 0.98     | 864     |

*Tabela 4.1 Resultados apresentados pelo modelo desenvolvido para descobrir o tipo de fatura*

Na Figura 4.1 apresentam-se três faturas processadas, cuja tipologia foi identificada corretamente e de acordo com os parâmetros da Tabela 4.1. Com base no modelo desenvolvido e nos resultados obtidos na classificação de faturas foi entendido que poderia ter uma utilização abrangente por ser um modelo que apresenta resultados fiáveis e consistentes. De realçar que o modelo foi aplicado a um universo de 3456 faturas para treino e 864 faturas para teste, sendo desejável que se continue a fazer o processamento de mais faturas a fim de validar se o comportamento do modelo se mantém ou

mesmo melhora para os indicadores em que a precisão não atingiu os 100%, nomeadamente no caso do Subsistence.

A Figura 4.1 apresenta três faturas em que a primeira pertence a uma fatura de restaurante e o modelo identificou o tipo Subsistence, a segunda é uma fatura da Uber, sendo categorizada no tipo Taxi e por fim a última fatura de um hotel foi considerada no tipo Accomodation.

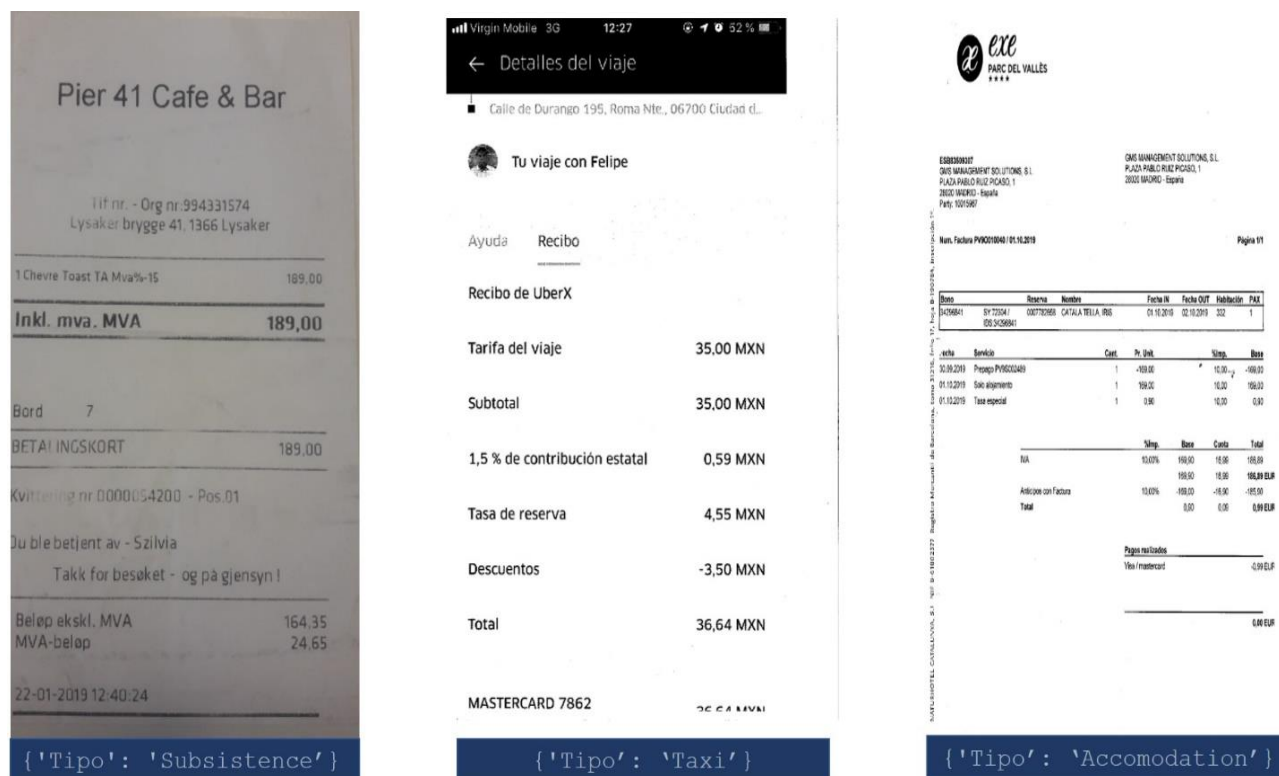


Figura 4.1: Três faturas com o respetivo Tipo identificado pelo modelo de ML

## 4.2 Resultados da solução com algoritmos de procura

Inicialmente, foi possível analisar cerca de 4.000 faturas variadas e desenvolveram-se algoritmos que generalizaram ao máximo a procura dos campos necessários com base em técnicas relacionadas com NLP, Deep Learning e Machine Learning. Nem sempre é possível utilizar estas técnicas para descobrir alguns campos específicos, pelo que foi necessário construir algoritmos com outros suportes cujo objetivo fosse procurar os mesmos.

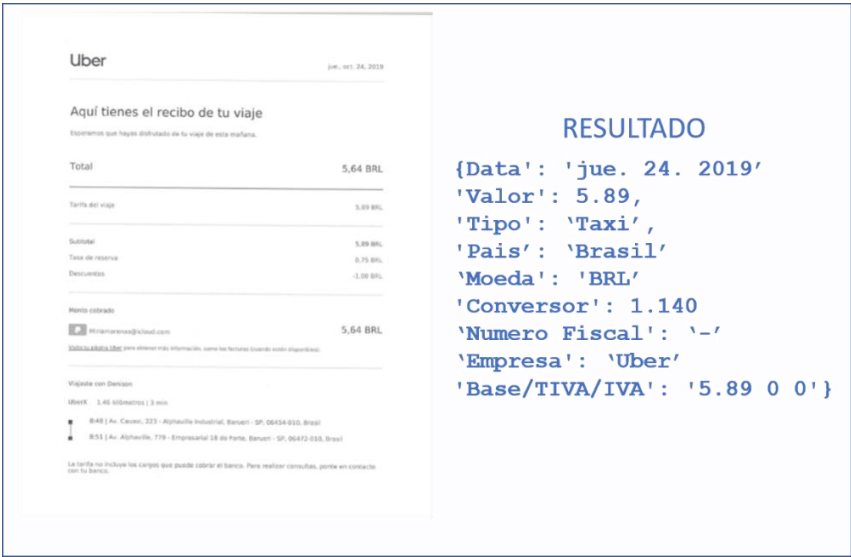
No caso específico da primeira solução pretendeu-se fazer a análise de resultados do processamento de 200 faturas espanholas. Para compreender estes resultados é necessário referir dois aspetos essenciais desta solução: não é escalável e o tamanho da amostra pode não ser conclusivo. A escalabilidade é de extrema importância neste trabalho porque o objetivo é conseguir identificar a maior

variedade de faturas e isso obtém-se ao testar grandes volumes de dados. Por outro lado, se a amostra é pequena poderá não permitir ter a certeza se a solução obtida com estes algoritmos será a mais eficaz.

Apresentam-se alguns exemplos de faturas de Espanha e uma do Brasil e mostram-se os resultados obtidos, após o que serão descritos os resultados finais relacionados com as 200 faturas espanholas testadas.

Na Figura 4.2 é apresentada uma fatura do Brasil que apresenta quase todos os campos certos, mas com algumas falhas. O valor total apresenta o primeiro erro e as causas devem-se, por um lado à formatação da fatura e por outro à incapacidade de resposta do algoritmo desenvolvido para fazer a procura do valor total. Pode ser visualizado que o valor total é 5.64, no entanto a aplicação identificou 5.89. Existiu este problema porque a fatura apresenta uma tarifa de 5.89, que é o valor superior, e desta forma, o algoritmo selecionou este campo como sendo o do valor total.

O número fiscal também não foi identificado porque cada país possui um tipo de número fiscal específico, por isso é necessário desenvolver algoritmos específicos para cada país. A primeira fase da aplicação apenas contém um algoritmo direccionado para procuras de números fiscais espanhóis, então para pesquisa deste campo relativamente a outros países ainda não é possível obter uma resposta com esta solução. Por fim, os valores de IVA não foram encontrados porque a fatura não apresenta a percentagem de IVA aplicável o que impediu o algoritmo de proceder aos cálculos de acordo com as parametrizações efetuadas.



The image shows a screenshot of an Uber receipt from Brazil on the left and its corresponding JSON result on the right. The receipt is in Portuguese and includes details such as the date (jue. 24. 2019), total amount (5,64 BRL), and various taxes and discounts. The JSON result on the right, titled 'RESULTADO', contains the following data: {Data: 'jue. 24. 2019', Valor: 5.89, Tipo: 'Taxi', Pais: 'Brasil', Moeda: 'BRL', Conversor: 1.140, Numero Fiscal: '-', Empresa: 'Uber', Base/TIVA/IVA: '5.89 0 0'}. The JSON result is displayed in blue text.

| Uber  |           |
|---|-----------|
| jue. 24. 2019   |           |
| Aquí tienes el recibo de tu viaje   |           |
| Dispositivos que hayan distribuido de tu viaje de esta mañana.  |           |
| Total   | 5,64 BRL  |
| Tarifa del viaje  | 5,89 BRL  |
| Subtotal  | 5,89 BRL  |
| Tasa de reserva   | 0,75 BRL  |
| Descuentos  | -1,00 BRL |
| Monto cobrado   | 5,64 BRL  |
| Visítanos en Uber para obtener más información, como los horarios de salida y llegada.                              |           |
| Viajaste con Destino  |           |
| UberX 3,40 kilómetros (3 min)   |           |
| 8481   Av. Casco, 223 - Alphaville Industrial, Barueri - SP, 06454-000, Brasil                                      |           |
| 8551   Av. Alphaville, 179 - Empresarial 18 de Junho, Barueri - SP, 06472-000, Brasil                               |           |
| La tarifa no incluye los cargos que puede cobrar el banco. Para realizar consultas, ponte en contacto con tu banco. |           |

```
{Data: 'jue. 24. 2019',
Valor: 5.89,
Tipo: 'Taxi',
Pais: 'Brasil',
Moeda: 'BRL',
Conversor: 1.140,
Numero Fiscal: '-',
Empresa: 'Uber',
Base/TIVA/IVA: '5.89 0 0'}
```

Figura 4.2: Primeira fatura com os respetivos resultados obtidos pela solução que utiliza algoritmos de procura pelos campos

A Figura 4.3 apresenta uma fatura espanhola com todos os campos detetados corretamente. Foi possível, através de uma API desenvolvida à medida ir fazer uma procura a um website das finanças e com base no número fiscal da empresa detetar o nome da empresa que corresponde a esse contribuinte fiscal. Esta API está programada para responder, atualmente, apenas a Espanha, podendo futuramente ser ajustada para poder dar idêntica resposta em outros países de acordo com a realidade específica de cada um deles.



*Figura 4.3 Segunda fatura com os respetivos resultados obtidos pela solução que utiliza algoritmos de procura pelos campos*

A Figura 4.4 é uma fatura espanhola e é possível identificar que o campo data não foi reconhecido corretamente. Este erro pode ter decorrido da Amazon Textract não identificar corretamente o texto ou a Amazon Comprehend não ter conseguido fazer a leitura correta da entidade data.





Seguidamente, na Tabela 4.2 são apresentados os resultados globais obtidos no tratamento da amostra de 200 faturas espanholas.

| <b>Campos</b>   | <b>Resultado</b> |
|-----------------|------------------|
| Valor Total     | 74%              |
| Data            | 95%              |
| Número Fiscal   | 64%              |
| País            | 100%             |
| Moeda           | 100%             |
| Conversor Moeda | 74%              |
| Empresa         | 55%              |
| Tipo            | 92%              |
| TIVA/IVA/BASE   | 65%              |

*Tabela 4.2 Resultados obtidos num conjunto de 200 faturas espanholas.*

No campo relativo ao Valor Total a percentagem de reconhecimento obteve 74%, dado que o algoritmo de procura encontrou dificuldades no processo: poderem existir valores inteiros ou decimais definidos como valor total, existirem valores superiores ao valor total tal como os IVAs, bem como alguns símbolos como o Euro serem identificados como valores pela Amazon Textract, entre outros.

O campo Data apresentou 95% de acerto na pesquisa, sendo o algoritmo de procura com melhores resultados, dado que a Amazon Comprehend possibilita uma procura precisa. O fator crítico na análise do campo Data é a normalização que o algoritmo requer no valor DD/MM/YY, pois existem múltiplas hipóteses na sua apresentação. Para ultrapassar esta dificuldade optou-se por considerar que sempre que exista um retorno plausível da data esta é considerada como um valor correto, daí o valor de precisão neste campo ser elevado.

O campo Número Fiscal apresentou uma taxa de correspondência de 64% e foi desenvolvido apenas para Espanha. Esta taxa de erro acontece uma vez que frequentemente se verifica que a Amazon Textract não identifica corretamente uma letra ou número, ocorrendo o retorno não correto do número fiscal.

Os campos País e Moeda apresentam o resultado de 100% pois a aplicação permite ao utilizador introduzir o input do país da fatura. Deste modo, a partir dessa informação, é possível retornar dados

como os IVAs do país e respetiva moeda, facilitando a procura dos restantes campos que se baseiam nos mesmos.

O campo Conversor de Moeda apresenta uma taxa de correspondência de 74%, estando ligado ao campo do valor total bem como ao país e à moeda. Sem a existência do valor total não é possível haver retorno correto da conversão para a moeda pretendida. As falhas de acerto estão diretamente associadas às falhas já identificadas acima na deteção do valor total.

O campo referente à deteção da Empresa apresentou uma taxa de acerto de 55% e tal como o número fiscal teve uma parte do algoritmo específica apenas aplicável a Espanha. Quando o número fiscal não é encontrado é utilizada a segunda parte do algoritmo em que é feita uma procura sobre a fatura, considerando o posicionamento de palavras específicas. Esta procura pode não ser totalmente eficaz e retornar valores que não correspondem ao nome da empresa.

O campo Tipo de fatura apresentou um resultado de 92%, que está de acordo com o descrito no ponto 4.1, de elevada precisão na deteção da tipologia de faturas. Como o modelo, apenas assentou o seu treino em 4.000 faturas é possível, por este facto, explicar o não alcance de 100%.

O campo TIVA/IVA/BASE alcançou uma precisão de 65% dado que está ligado com o valor total, percentagens do IVA que variam por tipologia de produto, bem como com os dados se apresentam nas faturas. A conjugação destas variáveis torna complexo o processo de deteção deste campo em algumas faturas, baixando a taxa de acerto.

### **4.3 Resultados da solução com modelo de Deep Learning**

Tal como foi referido no ponto 3.5.1 foi utilizado o modelo pré-treinado Faster-RCNN-Inception-V2, juntamente com a Tensorflow Object Detection API da Google. O modelo foi treinado com 2000 faturas e foram utilizados os parâmetros default apresentados pelo mesmo modelo. O principal benefício de utilizar um modelo pré-treinado é o conhecimento que possui, a partir de imagens que serviram de base para o seu treino. O modelo foi desenvolvido para prever os campos Valor Total e Data.

No dataset foram introduzidos dois tipos de conjuntos de imagens. Primeiro, um conjunto de imagens repetidas, mas com diferentes valores, por exemplo, faturas da Uber correspondentes a vários dias. No outro conjunto, introduziram-se faturas únicas, ou seja, faturas que não são repetidas, de forma a perceber se o modelo conseguiria identificar faturas desse tipo, treinando apenas com uma.

A avaliação do modelo resultou da análise de dois parâmetros, a perda total e a precisão que foi desenvolvida, com recurso à análise dos resultados obtidos com as faturas testadas.

A perda total foi calculada com recurso ao Tensorflow, dado que possibilita apresentar os resultados a cada iteração. A Figura 4.6 apresenta a perda total que o modelo foi tendo ao longo das várias iterações. O modelo apresentou inicialmente a perda total de valor 2 e ao fim de 2800 iterações este valor diminuiu para 0.18. A partir deste ponto a perda total começa a flutuar, à medida que vão sendo incrementadas as iterações, podendo isto representar o overfitting do modelo. Ao fim de 30.000 iterações foi possível identificar o momento em que o modelo alcançou o melhor resultado, neste caso ao fim de 10.000 iterações, alcançando o valor 0.03 para a perda total.



*Figura 4.6: Gráfico que identifica a perda total do modelo treinado em DL*

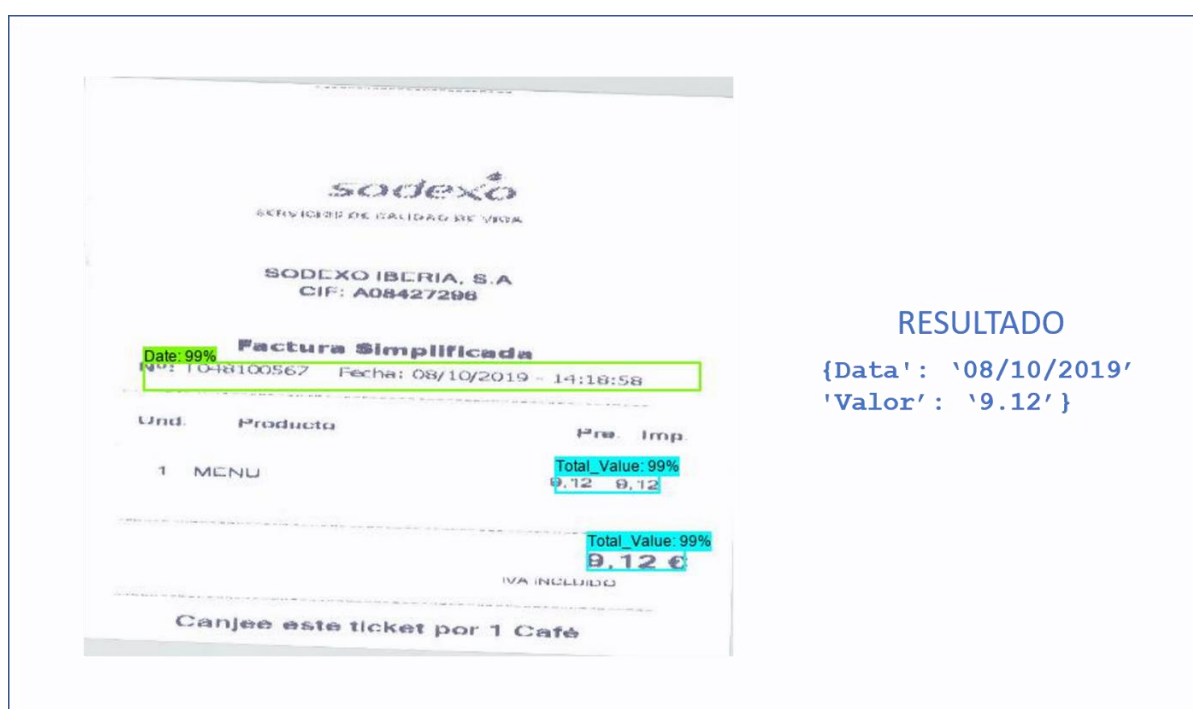
A precisão foi analisada de uma forma diferente, tendo sido feitos testes com múltiplas faturas e observados os resultados obtidos, permitindo deste modo concluir qual a precisão do modelo. Em deteção de objetos, a precisão tem em conta a posição das caixas. Neste modelo existe a possibilidade de identificar o valor correto com base na caixa mais próxima, tal como mencionado no ponto 3.5.2. Por isso, foi considerado para o parâmetro precisão, acertar no posicionamento dos limites da caixa onde se localizam os valores corretos. Nalguns casos, as faturas apresentam múltiplos Valores Totais ou Datas, pelo que se o modelo identificar algum deles corretamente, considera-se que foi preciso. A análise efetuada suportou-se em três grupos de faturas:

1. O primeiro grupo inclui múltiplas faturas que são semelhantes.
2. O segundo grupo inclui as faturas que são únicas.

3. O terceiro grupo inclui faturas que nunca foram vistas pelo modelo de treino.

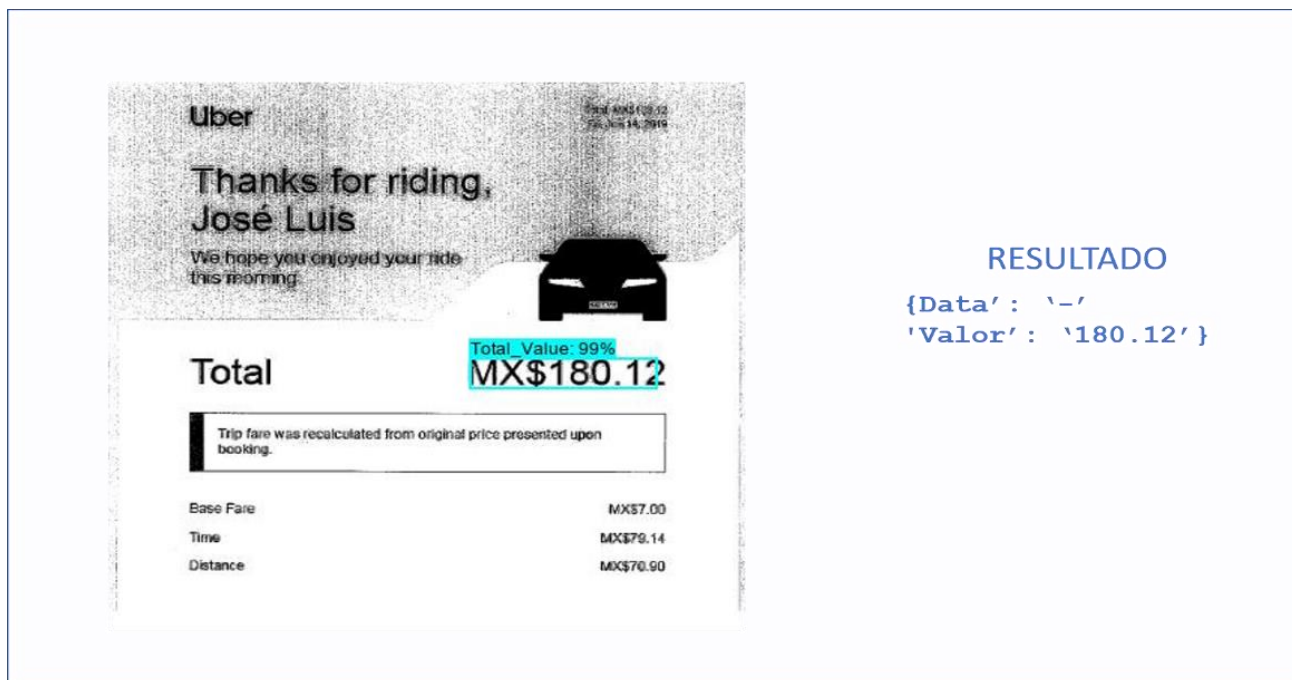
O primeiro grupo analisado apresentou resultados muito positivos. Pode-se concluir que o modelo identifica com precisão os resultados e com índices de confiança elevados. Nestas faturas destacam-se as empresas Sodexo e Uber que tinham um elevado número de faturas neste grupo de análise. De seguida, são apresentados exemplos e evidenciados os respetivos resultados das faturas que foram mais representativas a nível deste grupo.

Na Figura 4.7 é possível observar que o modelo conseguiu acertar nos dados desejados, ou seja, são retornadas as coordenadas corretas do Valor Total e Data.



*Figura 4.7: Primeira fatura com os respetivos resultados obtidos pela solução que utiliza o modelo de DL pertencente ao grupo 1*

Na Figura 4.8 foi apenas possível identificar o Valor Total. Isto ocorreu dado que a qualidade da imagem da Data não é compatível com o modelo de treino desenvolvido, e assim não foi possível detetá-la.



**RESULTADO**  
{Data': '-'  
'Valor': '180.12'}

Figura 4.8 Segunda fatura com os respectivos resultados obtidos pela solução que utiliza o modelo de DL pertencente ao grupo 1

No segundo grupo, o modelo foi treinado com uma amostra única de cada fatura. As Figuras 4.9 e 4.10 mostram que na primeira fatura os dados retornados não estão corretos e que, na segunda, não existe retorno de nenhuma resposta.

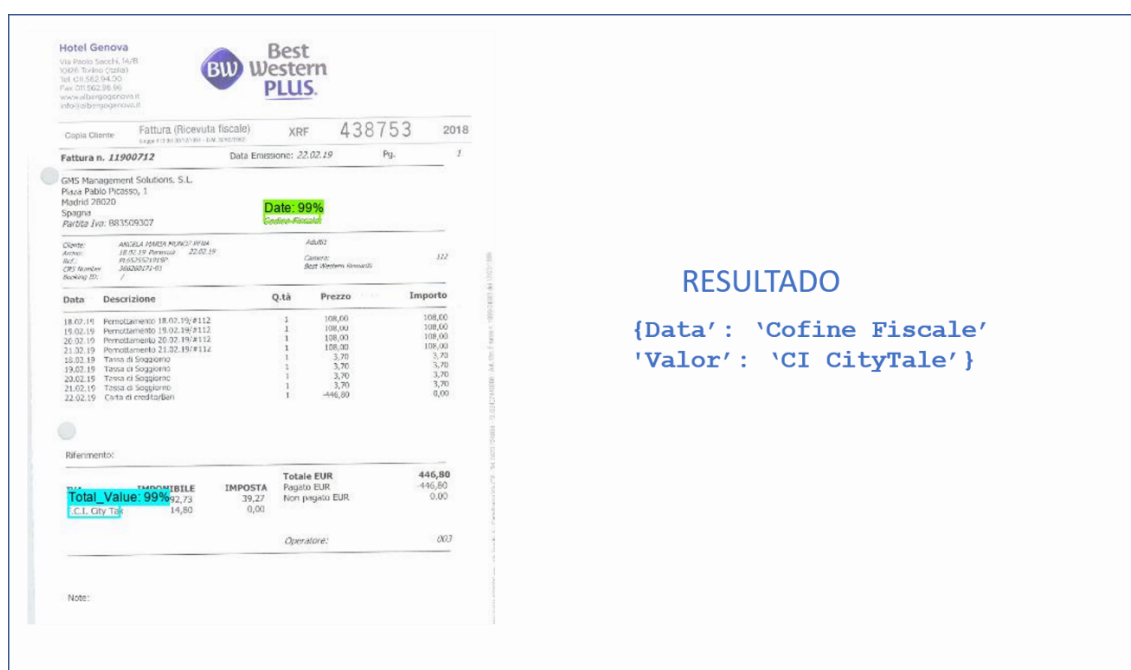




Figura 4.10 Segunda fatura com os respetivos resultados obtidos pela solução que utiliza o modelo de DL pertencente ao grupo 2

Por fim, no terceiro grupo foram testadas faturas em que o modelo não tinha sido treinado. Apesar do modelo não ter sido treinado, conseguiu detetar o valor total na Figura 4.11, algo que não parecia ser possível, dado que no grupo 2 os resultados obtidos não tinham sido positivos. Na Figura 4.12 não foi detetado nenhum dos campos pretendidos.

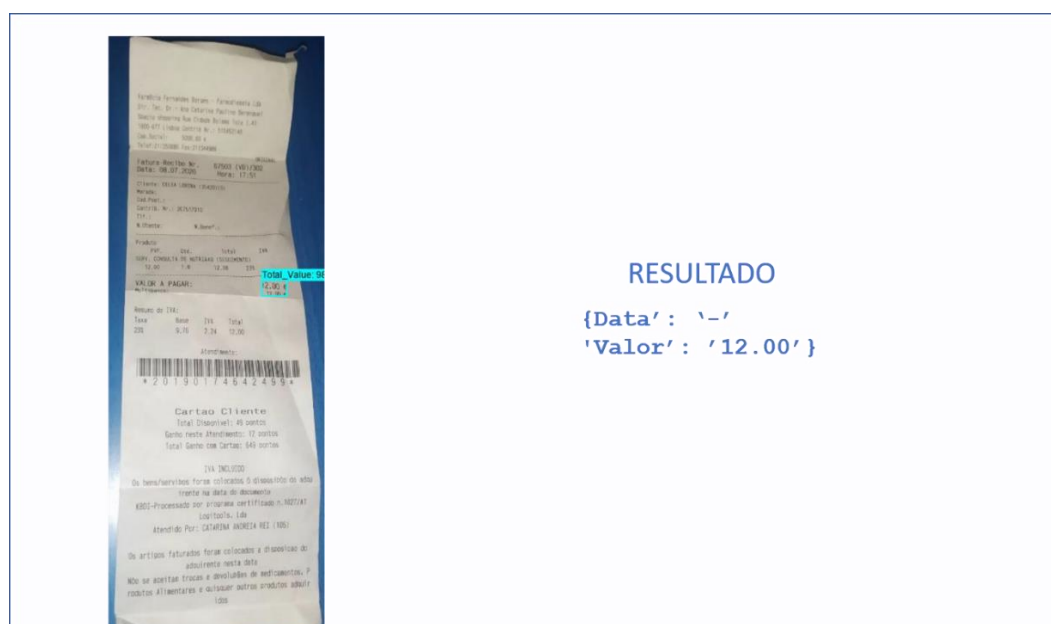


Figura 4.11 Primeira fatura com os respetivos resultados obtidos pela solução que utiliza o modelo de DL pertencente ao grupo 3

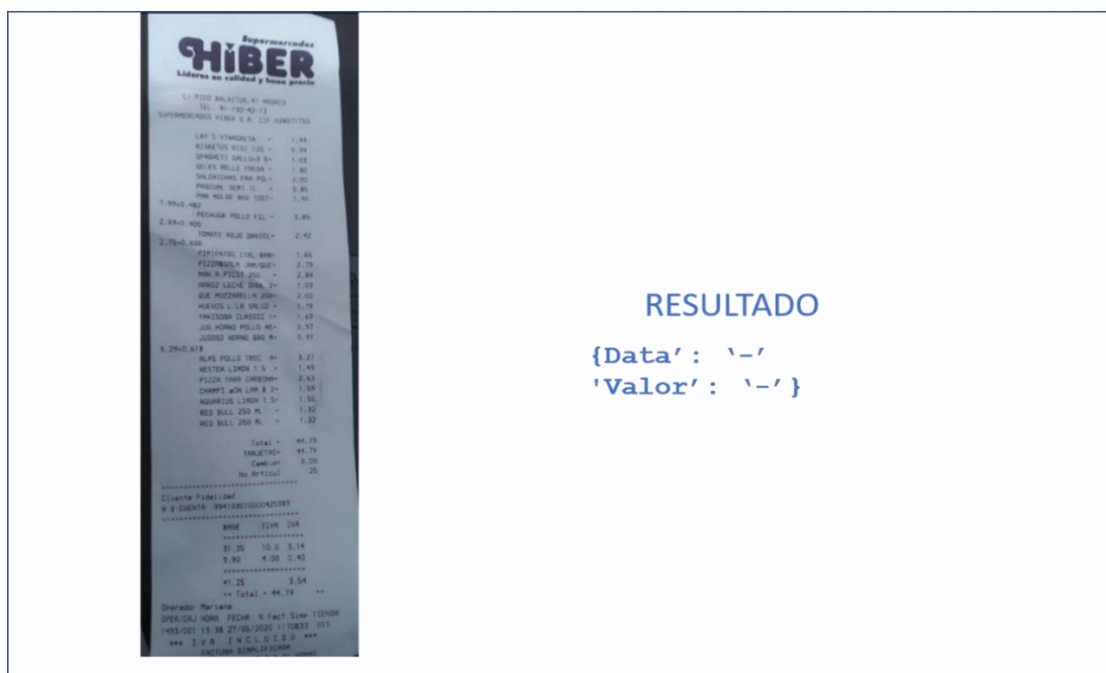


Figura 4.12 Segunda fatura com os respectivos resultados obtidos pela solução que utiliza o modelo de DL pertencente ao grupo 3

Quanto ao parâmetro da precisão, após testar as cerca de 440 faturas, pertencentes aos dados de validação, foram obtidos os resultados constantes na Tabela 4.3. Foram obtidos 78% e 70% para o Valor Total e Data respectivamente. Estes valores não são tão precisos como desejado, provavelmente por serem afetados pelas faturas únicas e pelas faturas não existentes no conjunto de treino. A precisão da Data é mais baixa que o Valor Total pois é um campo mais complexo e diversificado na forma de apresentação, o que dificulta a sua identificação.

| Campos      | Precisão |
|-------------|----------|
| Valor Total | 78%      |
| Data        | 70%      |

Tabela 4.3 Resultados da precisão com recurso ao modelo de DL

Neste projeto foram desenvolvidas duas soluções: uma baseada em algoritmos específicos desenvolvidos ou em serviços da Amazon que permitem fazer as identificações dos campos a descobrir e a outra que consistiu no desenvolvimento de um dataset que é atualizado de forma automatizada e num modelo de classificação que permite a identificação dos campos com base em bounding boxes



numa imagem. O objetivo pretendido era a utilização de uma solução que fosse melhorando com o tempo e apresentando resultados mais precisos e eficazes. A segunda solução desenvolveu-se em torno desse objetivo conseguindo alcançar os resultados apresentados na Tabela 4.3.

De facto, os resultados obtidos pela segunda solução foram muito encorajadores, mas importa referir que o dataset tem em torno de 2000 faturas. Seria necessário comprovar se com um aumento do dataset, se continuariam a obter os mesmos resultados ou até melhores. Para superar a dificuldade de não existir um volume elevado de dados de teste, utilizou-se um modelo pré-treinado, que ajuda o modelo de classificação a prever onde estão os campos corretos. Este modelo pré-treinado utilizou um tipo de imagens que não se assemelha às imagens a prever, por isso não existem factos que comprovem certezas relativamente a melhorias, se o modelo pré-treinado utilizasse dados semelhantes aos dados do dataset. Também não foi possível comprovar se, sem a existência do modelo pré-treinado mas com uma maior quantidade de faturas, os resultados continuariam positivos e melhores. Durante o processo observou-se um ponto negativo relativo à identificação incorreta dos campos, sempre que não existisse um número significativo de faturas do mesmo tipo.

É difícil comparar as duas soluções porque não existem certezas relativamente à segunda. Claramente, para uma fase inicial da melhoria do processo de tratamento de faturas a primeira solução apresenta melhores resultados, dado que consegue descobrir mais campos numa amostra maior, enquanto que a segunda solução alcança resultados apenas para tipos específicos que são treinados pelo modelo de classificação.

Em suma, é necessário fazer mais testes com um maior conjunto de dados e diferentes modelos de classificação, para se perceber se a segunda solução poderá ser posta em prática em produção. Deste modo, conclui-se que a primeira solução já foi testada e evidenciou resultados positivos para ser posta em funcionamento na empresa com segurança.

# Capítulo 5

## Conclusões e Trabalho Futuro

### 5.1 Conclusões

A MS solicitou o desenvolvimento de uma solução que permitisse otimizar o processo de gestão de faturas de despesas, que pudesse ter utilização a nível interno e, futuramente, poder ser incluída na oferta de serviços da empresa. Assim, no âmbito deste projeto foi criada uma solução contribui para a simplificação, incremento da eficiência e da eficácia do processo de tratamento de faturas.

A empresa definiu como princípio que deveria recorrer-se a uma plataforma de nuvem, utilizando os respetivos serviços. Após análise de várias opções selecionou-se a Amazon para o desenvolvimento deste projeto.

A solução inicia-se com a obtenção de uma foto de uma fatura, tirada por um telemóvel com um certo padrão de qualidade, são depois extraídos os campos desejados pela MS, tais como o país de emissão, a empresa que emitiu a fatura, a data, o valor total, os impostos associados, a moeda e as respetivas subdivisões, fazendo o respetivo armazenamento dos dados mais críticos e relevantes para a empresa.

Durante o processo foram identificadas duas abordagens uma de curto prazo que pudesse dar resposta imediata e outra de longo prazo que fosse escalável em função do volume de dados a processar. A primeira abordagem assentou numa solução que recorre a algoritmos que fazem procuras sobre os campos específicos, sendo uma solução para dar resposta imediata no arranque do projeto. A segunda abordagem consistiu numa solução de longo prazo, de um modelo de deep learning, que permite o treino à medida que vão sendo introduzidas mais faturas, sendo melhorada continuamente com o reconhecimento sucessivo de faturas novas e diferentes.

Para suportar o modelo de DL foi desenvolvido um dataset automatizado, que permite que à medida que o utilizador envia faturas e respetivas respostas, o modelo seja atualizado. Esta automatização, potenciou ganhos em termos de horas despendidas pelos colaboradores para carregamento dos dados e no posterior tratamento da informação.

Para cada uma das soluções foi necessário comprovar a sua capacidade de resposta, através de testes que foram realizados, a partir de uma amostra de faturas. Os testes iniciais permitiram analisar a tipologia de faturas, tendo sido constatados resultados positivos, na ordem de 98% de acerto do universo das 864 faturas testadas.

Foram ainda realizados testes às soluções para identificação dos campos que integram as faturas. Primeiramente, testaram-se 200 faturas espanholas com a solução assente nos algoritmos de procura.

Verificou-se que os campos de maior acerto corresponderam ao País, Moeda, Tipo e Data, dado que estes campos dão a possibilidade ao utilizador de validar ou mesmo registar diretamente o país correspondente. A Data contém um algoritmo de NLP suportado pelo Amazon Comprehend que apresenta retornos quase sempre corretos, permitindo a obtenção de taxas de acerto elevadas.

Por outro lado, o Valor Total, Empresa, Número Fiscal e TIVA/IVA/BASE, tiveram os piores resultados em termos de taxas de acerto porque dependem totalmente da validação do algoritmo de procura, não possuindo outras vias que facilitem a deteção de erros. Estes erros podem estar associados ao texto que é mal interpretado pelo Amazon Textract ou a fatura pode apresentar um formato em que o algoritmo não está preparado para fazer essa interpretação. Apesar destes constrangimentos conclui-se que os resultados foram positivos nesta solução.

Adicionalmente foram desenvolvidos testes para a solução que assenta num modelo de deep learning. Os testes incidiram em 2000 faturas com a avaliação a dois campos, Valor Total e Data, pretendendo-se aferir a precisão e a perda total do modelo. Com recurso à biblioteca Tensorflow foi possível treinar o modelo e obter de forma automática os resultados da perda total após cada iteração e fazer a análise dos respetivos valores.

Constatou-se que o modelo após 2800 iterações entrou num estado de overfitting, atingindo o melhor resultado de cerca de 0.03 de perda total. A precisão foi analisada com base em três grupos de faturas: o primeiro grupo inclui múltiplas faturas que são semelhantes, o segundo grupo inclui as faturas que são únicas e o terceiro grupo inclui faturas que nunca foram vistas pelo modelo de treino.

Estes grupos permitiram concluir as opções de maior e de menor precisão do funcionamento do modelo, sendo que o primeiro grupo obteve os melhores resultados. O segundo e terceiro grupos apresentaram muitas falhas e na maioria das vezes o modelo não conseguia identificar nenhum campo. Mesmo assim, no terceiro grupo existiram algumas faturas que o modelo conseguiu reconhecer campos, permitindo concluir que se o modelo contiver mais dados tudo indica que os resultados possam ser melhores.

Outra das conclusões decorrentes deste trabalho foi que este modelo de deep learning permitiu o reconhecimento de texto das faturas, apresentando resultados que à partida havia dúvidas da sua viabilidade, dado que os modelos que fazem deteção de objetos estão mais orientados para figuras e não para texto.

Também foi possível neste trabalho, pesquisar uma plataforma que providenciasse de forma acessível a maior parte dos serviços necessários para o desenvolvimento dos algoritmos, facilitando a construção dos processos para esta solução, neste caso a Amazon.

Em termos globais, os resultados deste projeto foram positivos, tendo-se conseguido alcançar os objetivos propostos na fase de arranque e acordados com a MS.

## **5.2 Trabalhos Futuros**

A experiência deste projeto permite identificar um conjunto de oportunidades, seja no desenvolvimento dos algoritmos, ou no modelo de deep learning que se detalham de seguida.

Desejavelmente a MS pretende utilizar a solução assente em deep learning para reconhecimento e tratamento das faturas. Todavia, como se viu ao longo deste trabalho, para esta solução ser eficaz em termos de resultados é necessário uma recolha e tratamento de um elevado volume de faturas, o que pode significar um período de tempo que pode ser longo. Assim, é necessário que as duas soluções convivam durante algum tempo.

Para permitir que as duas soluções funcionem em simultâneo é desejável que os algoritmos desenvolvidos para a solução de pesquisa de campos sejam melhorados. Sugere-se que exista uma maior pesquisa de campos para abranger mais países, pois isso permitirá a utilização alargada aos vários escritórios onde a empresa opera. Outra iniciativa poderia ser a construção de algoritmos específicos associados a cada país para permitir, por exemplo, o cálculo do número fiscal de forma automatizada, uma vez que cada país tem regras específicas.

Na solução com deep learning, uma das oportunidades identificadas está dependente da necessidade de continuar a alimentar o dataset com faturas novas, permitindo uma maior robustez e precisão em termos de resposta. Outra ação que se sugere é a de testar a identificação de outros campos para além do Valor Total e Data. Só desta forma será possível compreender se o modelo manterá ou alterará o comportamento a nível da precisão e de perdas totais.

A solução desenvolvida assentou na plataforma Amazon, mas verifica-se que existem outras plataformas com serviços semelhantes, havendo também a possibilidade de se avaliar o custo benefício dessas soluções alternativas comparativamente com a solução escolhida.

Deste modo, conforme descrito na metodologia inicial no Ponto 1.3, para além do descrito nos parágrafos anteriores, é importante fazer a implementação das soluções criadas. Isto passa por corrigir e afinar as soluções desenhadas, proceder ao arranque em produtivo em Espanha, divulgando o modo

de funcionamento da aplicação, esclarecendo dúvidas aos Colaboradores e posteriormente alargar o funcionamento da aplicação aos outros escritórios.

## Bibliografia

- Srivastava, P., & Khan, R. (2018). A Review Paper on Cloud Computing. *International Journal of Advanced Research in Computer Science and Software Engineering*, 8(6), 17.
- Bairagi, S. I., & Bang, A. O. (2018). *Cloud Computing : History , Architecture , Security Issues. March.*
- Varia, J., & Mathew, S. (2014). *Overview of Amazon Web Services (Survey Report). January*, 1–30. [http://media.amazonwebservices.com/AWS\\_Overview.pdf](http://media.amazonwebservices.com/AWS_Overview.pdf) Acesso em 2020-01-10
- Paper, W. (2015). AWS vs . Azure vs . GCP Determining Your Optimal Mix of Clouds. *Cloudyn*.
- Collier, M., & Shahan, R. (2016). *Fundamentals of Azure*.
- Krishnan, S. P. T., & Gonzalez, J. L. U. (2015). Building Your Next Big Thing with Google Cloud Platform. *Building Your Next Big Thing with Google Cloud Platform, January 2015*. The Google Cloud Plataform Difference, pp 3-12.
- Chollet, F. (2018). *Deep Learning with Python*.
- Dutta, P., & Dutta, P. (2019). Comparative Study of Cloud Services Offered by Amazon, Microsoft and Google. *International Journal of Trend in Scientific Research and Development*, Volume-3(Issue-3), 981–985. <https://doi.org/10.31142/ijtsrd23170>
- Azure. (2020). Services | Computer Vision Obtido de Azure | Computer Vision: <https://azure.microsoft.com/pt-pt/services/cognitive-services/computer-vision/> Acesso em 2020-05-10
- Azure. (2020). Services | Translator Obtido de Azure | Translator: [https://azure.microsoft.com/pt-pt/services/cognitive-services/ translator/](https://azure.microsoft.com/pt-pt/services/cognitive-services/translator/) Acesso em 2020-05-10
- Azure. (2020). Services | Text Analytics Obtido de Azure | Text Analytics: [https://azure.microsoft.com/pt-pt/services/cognitive-services/ text-analytics/](https://azure.microsoft.com/pt-pt/services/cognitive-services/text-analytics/) Acesso em 2020-05-11
- Azure. (2020). Services | Storage Explorer Obtido de Azure | Storage Explorer: <https://azure.microsoft.com/pt-pt/features/storage-explorer/> Acesso em 2020-05-13
- Azure. (2020). Services | API Management Obtido de Azure | API Management: <https://azure.microsoft.com/pt-pt/services/api-management/> Acesso em 2020-05-15
- Azure. (2020). Services | Cosmo DB Obtido de Azure | Cosmos DB: <https://azure.microsoft.com/pt-pt/services/cosmos-db/> Acesso em 2020-05-16
- Azure. (2020). Services | Azure Functions Obtido de Azure | Azure Functions: <https://azure.microsoft.com/pt-pt/services/functions/> Acesso em 2020-05-17

Azure. (2020). Services | Azure Virtual Machines Obtido de Azure | Azure Virtual Machines:  
<https://azure.microsoft.com/pt-pt/services/virtual-machines/> Acesso em 2020-05-17

Google. (2020). Cloud Computing Services | Cloud Vision Obtido da Google | Cloud Vision:  
<https://cloud.google.com/vision/> Acesso em 2019-11-08 Acesso em 2020-06-18

Google. (2020). Cloud Computing Services | Natural Language Obtido da Google | Natural Language:  
<https://cloud.google.com/natural-language> Acesso em 2020-06-18

Google. (2020). Cloud Computing Services | Cloud Storage Obtido da Google | Cloud Storage:  
<https://cloud.google.com/storage> Acesso em 2020-06-19

Google. (2020). Cloud Computing Services | Cloud Endpoints Obtido da Google | Cloud Endpoints:  
<https://cloud.google.com/endpoints> Acesso em 2020-06-19

Google. (2020). Cloud Computing Services | Google Datastore Obtido da Google | Google Datastore:  
<https://cloud.google.com/datastore> Acesso em 2020-06-20

Google. (2020). Cloud Computing Services | Cloud Translation Obtido da Google | Cloud Translation:  
<https://cloud.google.com/translate> Acesso em 2020-06-20

Google. (2020). Cloud Computing Services | Cloud Functions Obtido da Google | Cloud Functions:  
<https://cloud.google.com/functions> Acesso em 2020-06-20

Google. (2020). Cloud Computing Services | Google Kubernetes Obtido da Google | Google Kubernetes:  
<https://cloud.google.com/kubernetes-engine> Acesso em 2020-06-20

Amazon. (2020). Services | Amazon Textract Obtido da Amazon | Amazon Textract:  
<https://aws.amazon.com/pt/textract/> Acesso em 2020-01-10

Amazon. (2020). Services | Amazon Rekognition Obtido da Amazon | Amazon Rekognition:  
<https://aws.amazon.com/pt/rekognition/> Acesso em 2020-01-12

Amazon. (2020). Services | Amazon Sagemaker Obtido da Amazon | Amazon Sagemaker:  
<https://aws.amazon.com/pt/sagemaker/> Acesso em 2020-01-14

Amazon. (2020). Services | Amazon S3 Obtido da Amazon | Amazon S3:  
<https://aws.amazon.com/pt/s3/> Acesso em 2020-01-16

Amazon. (2020). Services | Amazon API Gateway Obtido da Amazon | Amazon API Gateway:  
<https://aws.amazon.com/pt/api-gateway/> Acesso em 2020-01-17

Amazon. (2020). Services | Amazon Dynamo DB Obtido da Amazon | Amazon Dynamo DB:  
<https://aws.amazon.com/pt/dynamodb/> Acesso em 2020-01-18

Amazon. (2020). Services | Amazon Translate Obtido da Amazon | Amazon Translate:  
<https://aws.amazon.com/pt/translate/> Acesso em 2020-01-19

Amazon. (2020). Services | Amazon Lambda Obtido da Amazon | Amazon Lambda:  
<https://aws.amazon.com/pt/lambda/> Acesso em 2020-01-20

Amazon. (2020). Services | Amazon EC2 Obtido da Amazon | Amazon EC2:  
<https://aws.amazon.com/pt/ec2/> Acesso em 2020-01-21

Tensorflow. (2020). Documentation | Tensorflow:  
<https://github.com/tensorflow/docs> Acesso em 2020-05-19

Demeester, T., Rocktäschel, T., & Riedel, S. (2016). Lifted rule injection for relation embeddings.  
EMNLP 2016 - Conference on Empirical Methods in Natural Language Processing,  
Proceedings, 1389–1399.

Management Solutions, (2019). Website | History:  
<https://www.managementsolutions.com/pt-br> 2019-09-28